



TREBALL FINAL DE GRAU



ESCOLA
POLITÈCNICA SUPERIOR
UNIVERSITAT DE LLEIDA
INSPIRING THE FUTURE

Estudiant: Aleix Lladós

Titulació: Doble Grau en Enginyeria Informàtica i ADE

Títol de Treball Final de Grau: **Design and implementation of the application for the Lleida Loyalty Cross Program**

Director/a: **Santi Martínez i Jaume Codina**

Presentació

Mes: Juny

Any: 2018

INDEX

1. INTRODUCTION	5
2. THE AIM OF THE PROJECT	7
3. CUSTOMER LOYALTY	8
3.1. MEASURING CUSTOMER LOYALTY	8
3.2. LOYALTY MARKETING	9
3.3. LOYALTY PROGRAMS	10
4. THE SURVEY	13
4.1. THE RESULTS	17
4.2. SURVEY CONCLUSIONS	22
4.3. APPLICATION TO THE PROJECT	22
5. IMPLEMENTATION	24
5.1. THE DATABASE	24
5.1.1. DESIGN OF THE DATABASE	24
5.1.2. CLASSES	27
5.1.3. RELATIONSHIPS	29
5.2. IMPLEMENTATION OF THE DATABASE	31
6. THE SERVER	33
6.1. REGISTER	34
6.2. SELECT DISCOUNT TYPE	34
6.3. LOGIN	34
6.4. MODIFY ACCOUNT	35
6.5. ADD DISCOUNT TYPE	35
6.6. CLOSE ACCOUNT	35
6.7. LOCK ACCOUNT	35
6.8. TAKE VOUCHER	35
6.9. GIVE VOUCHER	36
6.9.1. GENERATING A VOUCHER	36
6.10. SELECT SECTOR TYPE	38
6.11. ADD SECTOR TYPE	38
7. MYSQL INJECTION	39
8. THE WEB APPLICATION	40

8.1. TESTING THE WEB APPLICATION.....	48
9. FUTURE PROPOSALS	56
10. CONCLUSIONS	57
11. BIBLIOGRAPHY	59

IMAGES INDEX

Image 1 Shop home page	40
Image 2 Shop Take Voucher	41
Image 3 Shop Add discount type	42
Image 4 Customer home page	42
Image 5 Customer Use voucher	43
Image 6 Customer Get voucher	44
Image 7 Customer Add sector type	44
Image 8 Shop Login	45
Image 9 Shop Register	45
Image 10 Shop Voucher's List	45
Image 11 Modify shop account	46
Image 12 Customer login	46
Image 13 Customer register	47
Image 14 Customer Voucher's List	47
Image 15 Customer Modify Account	47
Image 16 Lock Account	48
Image 17 Close Account	48
Image 18 Take/Give Voucher	50
Image 19 Take/Give Voucher Successful	50
Image 20 Register User	51
Image 21 Login Successful	51
Image 22 User Login	52
Image 23 Shop Login	52
Image 24 Login successful	52
Image 25 Modify Account	53
Image 26 Successful	53
Image 27 Warning	54
Image 28 Add Discount Type	54
Image 29 Successful Page	55
Image 30 Add Sector Type	55
Image 31 Successful Page	55

GRAPHICS INDEX

Graphic 1: Percentage of shops offering a discount	17
Graphic 2: Percentage of shops willing to offer a discount	18
Graphic 3: Discount type	18
Graphic 4: Quantity of percentage discount	19
Graphic 5: Quantity of amount discount.....	20
Graphic 6: Minimum quantity to offer a discount	20
Graphic 7: Minimum quantity to apply a discount	21

1. INTRODUCTION

Before starting, I would like to say that this project wouldn't have been possible without some people's help.

First of all, I would like to thanks Santi Martínez and Jaume Codina. They have helped me a lot with the different meetings that we have had. They have oriented me with everything I asked for. They have also provided the information needed in order to make the project.

Besides, I would also like to express my gratitude to my family, Xavier, Rosa and Oriol. They have been near me during the entire project and they have also helped me by passing the surveys to the different shops.

Finally, I would like to thanks my girlfriend, Berta, who has been very patience with me during the project and has helped me with everything she could.

A project of this size can't be done by only a person, and I think this project is the result of their collaboration, so, one more time, thank you!

Nowadays, we live in a society that is totally digitalized, not only in every job, but also in our day to day life. In the recent past, this society started the era of digitalization.

First, it appeared the television, then computers, mobiles, tablets, etc, a whole range of products that involved informatics. With these products, applications were born. An application is a computer program designed to perform a group of coordinated functions, tasks, or activities for the benefit of the user. Applications have made our life easier and more sophisticated. Applications have helped us to manage our life, and improve our business. Nowadays, we can find thousands of applications in the market, which can be used in thousand of different sectors.

In my opinion, our future is based on technology and on the applications. That's the reason why I chose this project as a final grade project. I wanted to see how the real process of creating an application is, how much work is there behind an application, how is it designed, and how is it implemented. In this project, I have only seen the part of designing and creating the database and the web application. It really motivated me, and finally I decided to do it.

During the degree, I have seen lots of programming languages, but I also wanted to learn a new one. I found the opportunity to learn a new one by doing this project, and that's why I have implemented the application in PHP and HTML.

On the other hand, I found it interesting to make a market analysis so as to improve the application. I thought that it could be interesting to let the local shops of Lleida express their opinion about a product that involves them. I wanted to know their opinion about the different kind of discounts that they offer in their shop or that they are willing to offer. I think this is an important point of the project, because the entire project is based on offering discounts, but, what kind of discounts do the shops prefer to offer? Are these discounts profitable to them?

Doing a final grade project has been the opportunity to apply all the knowledge acquired during the double degree. So, to sum up, I find very interesting doing a final grade project, as it is the best way to see what a project of this size is, and the work that there is behind it.

This project is composed of two parts. The first part is focused on the business side and the second part is focused on the engineering side.

In the first part, we will see a global and theoretical explanation of what customer's loyalty is. Besides, we are going to explain the results of the surveys that we have done so as to know the opinion of the different shops in Lleida.

On the other hand, the second part is totally focused on the implementation of the web application, applying the results obtained in the first theoretical part.

2. THE AIM OF THE PROJECT

This project is the continuation of the final grade project of Jorge Belluscio Portugues. It is the final grade project for “Universitat de Lleida”. It involves both degrees, Informatics engineering and administration and direction of business. This project is a part of the whole project belonging to UDL with Lleida town hall collaboration. The first phase of the project, developed by Jorge, was exclusively focused on “Requirements Engineering”.

The aim of the whole project is to create a customer loyalty application for the small business of Lleida, so as to improve the customer’s loyalty and help the small business to increase their market share. This project is based on a loyalty cross program.

In this second part we have made some updates to the first part, so as to make the implementation of the application easier. This part is focused on the design and implementation of the application, following the requirements previously acquired. It is composed by all the parts, the design and implementation of the database, the implementation of the server, and also the implementation of the web application. By now, as it is a prototype, it has only been implemented the web application.

If this project continues, the next step will be transforming this web application into an android application so as it can be used either in tablets or mobile phones. The idea of the project, as we said before, is to create a loyalty application for the small business of Lleida.

Nowadays, loyalty applications are quite useful, and usually the customers are very pleased with them. It is a good way to increase the customer’s loyalty and also increase the market share of the small business in a city.

Besides, this project will give another point of view for the application, as it has a direct contact with the small business of Lleida to have more information about what do they think about the discounts, what kind of discounts are better to them, which quantity, etc.

3. CUSTOMER LOYALTY

The customer loyalty is defined as: Likelihood of previous customers to continue to buy from a specific organization. Great attention is given to marketing and customer service to retain current customers by increasing their customer loyalty. (Business Dictionary, 2018) In my opinion, the customer loyalty is directly related with the grade of satisfaction that the customer has with a shop or brand.

At present, markets are characterized by a growing complexity and sophistication of customers, who are increasingly more educated and have more knowledge of the products, and are more demanding in their demands (Blackwell, Miniard and Engel, 2002, Alonso and Grande, 2004). Nowadays, due to the big progress that the world is experimenting, every customer has infinity varieties when he wants to purchase a product. There are a lot of enterprises, and a lot of competence. Every day, it is more difficult to have a customer's loyalty and that's why customer loyalty and loyalty techniques are so important. Nowadays, the post-buy is as important as the purchase itself. We want to receive a good treatment and the best rewards after the buy. That's what will make us go back to that shop.

3.1. MEASURING CUSTOMER LOYALTY

Loyalty means a customer wants to do business with the enterprise and does. The challenge for organizations is taking this definition and translating it into actual practice, where specific actions are defined and ideal customer relationships are envisioned. It also means identifying a means for taking this loyalty construct and putting it into measurable terms so that success and failure can be assessed and progress or decline tracked. (Ranade, 2012)

Customer loyalty can be measured and monitored. Loyal customers believe the products and services purchased from their supplier are superior to those of the competitors. Customers often think that it exist a relationship bigger than just the products that they are buying. Measuring loyalty means measuring the strength of this relationship between buyer and seller, between the organization and its customer. It is challenging to measure the level of customer loyalty within the relationship, which is why companies so often succumb to simply defining loyalty as the number of purchases made or a continuing pattern of buy behavior.

A common mistake of the enterprise is to measure customer loyalty by asking the customers whether they are loyal to the brand or not. Every customer would answer yes to that question to many enterprises at the same time.

What is important here is to measure the attitudes and the behavior of the customer, as we know that make up this concept of loyalty. For instance, we can focus on these seven behaviors:

- Likelihood that the customer recommends the products and services of the enterprise to others.
- Likelihood that the customer continues purchasing the products and services of the enterprise, minimum, at the same level.
- Likelihood that the customer purchases other products and services that the enterprise offers.
- Expect that the customer believes that the products of the enterprise are better than the others offered in the market.
- The customer does not seek alternative providers to replace yours.
- Providing the company with opportunities to correct problems and not using these as a basis for compromising the relationship.

In this project, we won't be able to measure the customer loyalty due to the application. However, in the future, it would be rather interesting to do an investigation about the results of the application, so as to correct the errors and improve the application.

3.2. LOYALTY MARKETING

Loyalty marketing is an approach to marketing, based on strategic management, in which a company focuses on growing and retaining existing customers through incentives. Branding, product marketing and loyalty marketing all form part of the customer proposition – the subjective assessment by the customer of whether to purchase a brand or not based on the integrated combination of the value they receive from each of these marketing disciplines

The discipline of customer loyalty marketing has been around for many years, but expansions from it merely being a model for conducting business to becoming a

vehicle for marketing and advertising have made it omnipresent in consumer marketing organizations since the mid to late 1990s. Some of the newer loyalty marketing industry insiders, such as Fred Reichheld, have claimed a strong link between customer loyalty marketing and customer referral. In recent years, a new marketing discipline called "customer advocacy marketing" has been combined with or replaced "customer loyalty marketing." To the general public, many airline miles programs, hotel frequent guest programs and credit card incentive programs are the most visible customer loyalty marketing programs. (Reichheld & Teal, 2001)

Usually, marketing strategies are designed by merchants so as to encourage customers to continue buying in the same shop. These kinds of strategies are used in small business. Due to the importance of loyalty marketing, it is very interesting to create an application that can meet the needs. Organizations employ loyalty programs which reward customers for repeat business.

3.3. LOYALTY PROGRAMS

Loyalty programs are structured marketing strategies designed by merchants to encourage customers to continue to shop at or use the services of businesses associated with each program. These programs exist covering most types of commerce, each one having varying features and rewards-schemes.

This project consists in a loyalty cross program. A loyalty cross program is a loyalty program that crosses the enterprises. An enterprise does not offer directly advantages to the customer, but to other enterprises. Then, a customer going to another enterprise will receive advantages to use in your enterprise. It is a crossed loyalty program.

Doing a cross program we get two advantages. The first one is that the enterprises get more loyalty from the customers. The second one is that customers are encouraged to keep buying products in the enterprises using this application. This application is focused on one marketing technique, which is very useful so as to improve the customer's loyalty.

Rogers and Shoemaker (1971) define the concept of propensity to innovate as "the extent to which an individual is relatively pioneering in adopting new ideas in relation to other members of the social system ». Nowadays, every business is "forced" to innovate. This application is the opportunity to create a loyalty program with technology.

The results of the research *The influence of managers' innovativeness on the implementation of loyalty programs in the small retailing sector* (Rodríguez-Del-Bosque and co., 2008), raise interesting implications for the management of small business. Specifically, a deep understanding of the factors that determine the acceptance of customer loyalty programs by small businesses is essential to define strategies and actions of modernization and revitalization of the sector in front of the big brands commercial.

It is essential to transmit to the owners and managers of small business companies the advantages of the implementation of loyalty programs they have for their business, not only in terms of customer loyalty but also in sales, cross-transactions or availability of management information, among others.

It is worth highlighting the differences in the process of adoption the customer loyalty programs among single-person enterprises and micro-enterprises. In this way, in the case of entrepreneurs who manage their business individual, reducing the perception of complexity in the implementation of programs of loyalty become particularly important. Also, for this group it is especially important advice and support in administrative and commercial management, which allows overcome the barriers derived from the scarcity of organizational resources. For his part, in the case of microenterprises with between 2 and 9 employees, it is essential to transmit them the impact of loyalty programs on terms of results, since the availability of resources allows them to face the implementation even if they are perceived as complex to manage.

This application is a loyalty program which is based in offering discounts to the customers. I think that the best way to win customer loyalty is to offer advantages to them when they want to buy some products. It consists in offering a discount based on the quantity that they have spent in the buy offer, so, the more they spent, the higher

is the discount. As we said before, it is a loyalty cross program. By this, customers will be encouraged to buy in the small business that has the application, and this small business will increase their sales.

Finally, after this research, we realized that we needed the implication of the small business so as to continue implementing the application. That's why we decided to create a survey so as to let the small business express their opinion about the discounts being offered in the application.

4. THE SURVEY

First of all, we have to say that in the project before, there was already a survey done to different shops in Lleida. This survey was focused on the whole application. With this survey, our goal is to focus on what kind of discounts the shops prefer to offer and the quantity of these discounts that they are willing to offer. Besides, we are also asking for the sector of the shop, so as we can relate the kind of discounts with the sector type of the shop.

As we said before, we have created a survey so as to let the shops express their opinion. When a new application is created, it is important to know the needs of the market. By doing a market analysis, we will be able to know in what kind of discounts we have to focus more. We will also know what kinds of discounts are more profitable for the shops in every sector. With this survey, we want to know if the shops currently offer any kind of discount, and if they are willing to offer some. We will also know what kind of discounts they prefer to offer, and the quantity of the discounts. This information will be very useful when implementing the application, because we will know in which kind of discounts we have to focus more, and the value of this discounts. Here we have the completely survey.

Discount's Survey

This survey will be profitable for the final degree project done in the University of Lleida so as to know the opinion of the little business and know which are their preferences about eh discounts.

*Obligatorio

1.- Which is the sector of your shop? *

- ☐ Hair Saloon
- ☐ Bars
- ☐ Jewelry
- ☐ Clothes
- ☐ Veterinarian
- ☐ Furniture
- ☐ Informatics
- ☐ Otro: _____

2.- Do you offer any kind of discount? *

☐ YES

☐ NO

2.1.- Are you willing to offer some, or continue doing it? *

☐ YES

☐ NO

3.- What kind of discount do you think best fits with your shop?
(More than one can be selected)

☐ Percentage Discount(Ex: 10%)

☐ Amount Discount (Ex: 5€)

☐ Discount for multiple product (Ex: 2x1 3x2)

☐ Discount for multiple buys

☐ Seasonal Discount

☐ Otro: _____

4.- In case of percentage, which value would you like to offer?
(More than one can be selected)

- ☐ 0-5%
- ☐ 5-10%
- ☐ 10-15%
- ☐ 15-20%
- ☐ +20%

4.1.- In case of amount, which value would you like to offer?
(More than one can be selected).

- ☐ 0-5€
- ☐ 5-10€
- ☐ 10-15€
- ☐ 15-20€
- ☐ +20€

5.- When a customer makes a purchase in your shop, is there a minimum quantity he has to spend so as to offer him a discount?

- ☐ There is no minimum quantity
- ☐ 0-10€
- ☐ 10-20€
- ☐ 20-30€
- ☐ 30-40€
- ☐ +40€

6.- When a customer makes a purchase in your shop, is there a minimum quantity he has to spend so as to apply him a discount?

- ☐ There is no minimum quantity
- ☐ 0-10€
- ☐ 10-20€
- ☐ 20-30€
- ☐ 30-40€
- ☐ +40€

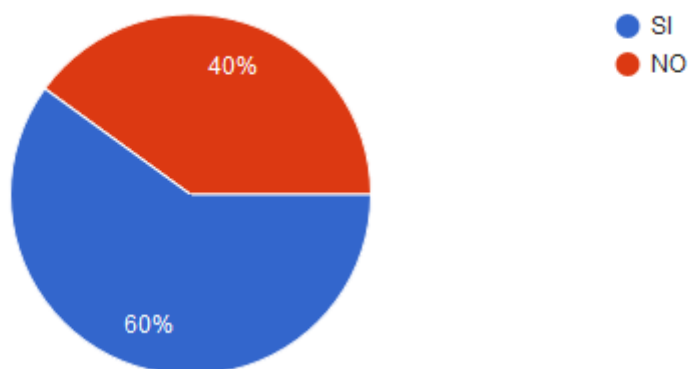
4.1. THE RESULTS

Here we are going to show the results after having passed the surveys for the different small business of Lleida. We have passed this survey in lots of different sector type's so as to know their opinion. We have done 110 surveys. Here are the results.

Question 1: Which is your sector type?

The first question asked about the sector type of the shop, and we have more than 20 different kinds of sectors.

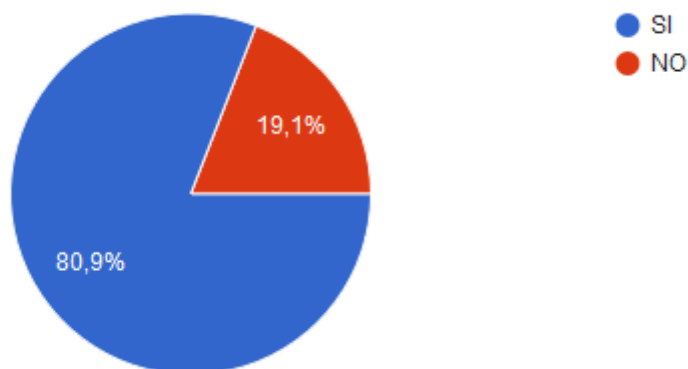
Question 2: Do you offer any kind of discount?



Graphic 1: Percentage of shops offering a discount

Nowadays, as we can see in this graphic, the 60% of the shops currently offer some type of discount. There's a 40% who does not offer any kind of discount.

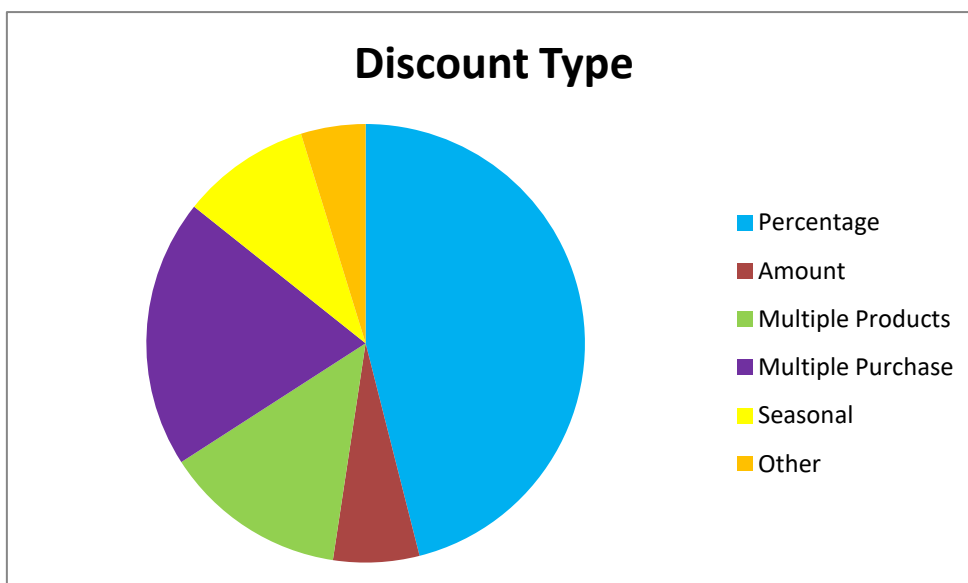
Question 3: Are you willing to offer some, or continue doing it?



Graphic 2: Percentage of shops willing to offer a discount

Here we can see a difference between the question asked before. We can see that the percentage of shops who want to offer a discount has increased by a 20% and now the 80% of the shops are offering a discount or are willing to offer one in the future. There's only the 19,1% who doesn't want to offer a discount.

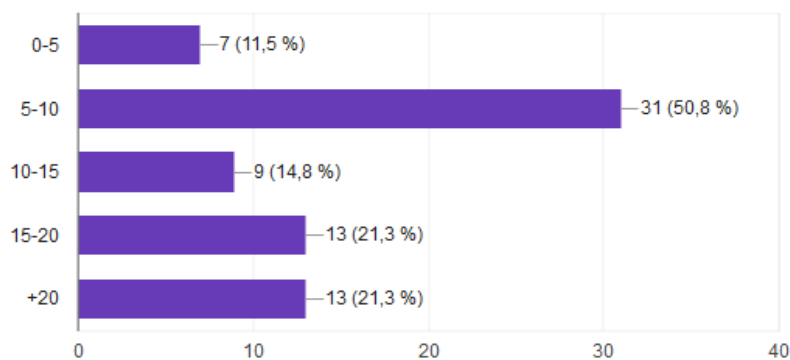
Question 4: What kind of discount do you think best fits with your shop?



Graphic 3: Discount type

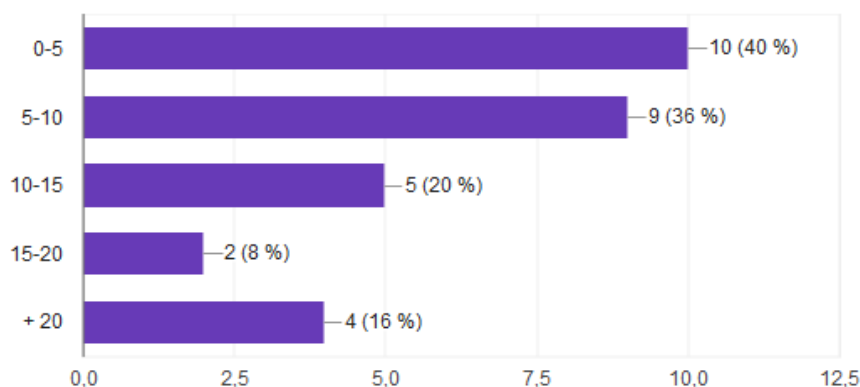
In this graphic, we can see the different discounts that the shops have chosen. We can see that with a 46% of the total, the percentage is the discount type that the shops prefer. The second type of discount chosen is the multiple purchases, for example giving a discount when the customer has bought five times in the shop. The option other was chosen only the 4% of the times, and the options were different such as depending on the provider.

Question 5: In case of percentage, which value would you like to offer?



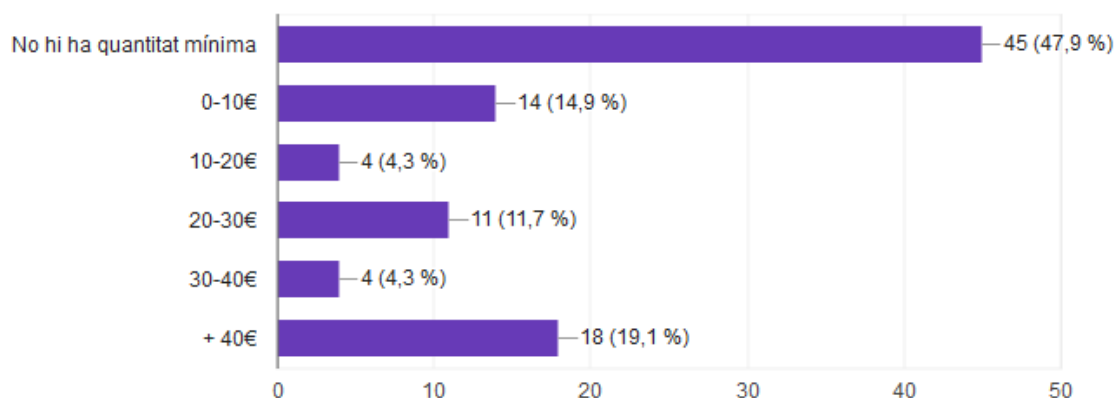
Graphic 4: Quantity of percentage discount

In this question, we were asking the quantity of percentage that the shops are willing to offer. We can see in the graphic that the most part of the shops, a 50,8%, would like to offer a percentage discount of between 5 and 10 %. The rest of the shops, would like to offer more than a 10% and there is a 21,3 % of the shops who would like to offer more than a 20% of discount. We can see that all the percentages are more than the 100% and that's because there was more than more than one option as a possible response.

Question 6: In case of amount, which value would you like to offer?

Graphic 5: Quantity of amount discount

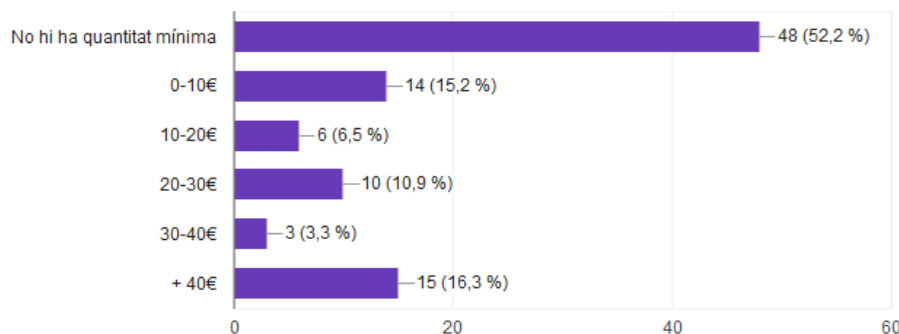
In this question we were asking the quantity of the discount that the shops are willing to offer in case they chose before the option of amount. We can see that the most part of the shops would like to offer a discount of between 0€ and 5€. Besides, we can see that there are only a 16% of the shops who would like to offer more than 20€ of discount and only an 8% of the shops would like to offer between 15€ and 20€. So, we can say that the shops would like to offer between 0 and 15€. We can see that all the percentages are more than the 100% and that's because there was more than more than one option as a possible response.

Question 7: When a customer makes a purchase in your shop, is there any minimum quantity he has to spend so as to offer him a voucher?

Graphic 6: Minimum quantity to offer a discount

In this graphic, we can see that the most part of the shops, almost a 50%, considers that there is no minimum quantity that a customer has to spend in the shop so as to offer him a discount. So, for instance, if a customer spends 1€ in the purchase, he would also receive a discount for another purchase. Otherwise, there are almost a 20% of the shops who considers that this quantity must be more than 40€. After having analyzed the results, we have seen that the shops who have chosen the option of more than 40€ have something in common. We have observed that the type of shops are shops who the products that they sell are more expensive that the other. For example, we are talking about sector type's like jewelry, mechanic, etc. We can see that all the percentages are more than the 100% and that's because there was more than more than one option as a possible response.

Question 8: When a customer makes a purchase in your shop, is there a minimum quantity he has to spend so as to apply him a discount?



Graphic 7: Minimum quantity to apply a discount

The results of this question are quite similar to the question before. In this case, we were asking if there is any minimum quantity that a customer has to spend in a shop so as to apply him a discount. We can see that the 52% of the shops say that there is no minimum quantity. and the 15% says that the quantity must be between 0€ and 10 €. Otherwise, we can see that there is a 16% of the shops that say that the quantity must be more than 40€ so as to apply a discount. We can see that all the percentages are more than the 100% and that's because there was more than more than one option as a possible response.

4.2. SURVEY CONCLUSIONS

In conclusion, we have realized that this survey has been very useful so as to know the opinion of the shops. We have seen that currently there is only the 60% of the shops that are offering discounts, but there is an 80% that are willing to offer some. So, with an application like this that we are creating, this kind of shops that are willing to offer discounts will have the perfect opportunity so as to start offering some. Besides, we have seen that the best kind of discount type for the shops is the percentage type. However, we have also seen that for certain kind of shops, the discount types that they offer are quite different. For example, in the case of bars, the best discount type is for multiple purchases, like offering a discount after the customer has done 5 purchases in the bar.

In the case of the quantity of the discounts, we have seen that the most part of the shops prefer to offer a discount of between 0€ and 10€ or 0% and 10%. However, there are some shops that would like to offer a quantity bigger than 20.

We have also seen that the most part of the shops does not have a minimum quantity in a purchase so as to offer or apply a discount. However, we have realized that there are certain types of shops who would like to have a minimum quantity bigger than 40€ so as to offer or apply a discount. This kind of shops, as we have said before, are those selling a product or offering a service which the cost is quite high. Then, so as to offer a discount, the money spent by a customer in a purchase must be bigger than other shops.

Finally, we can say that this survey has been very useful so as to implement and take decisions about the application. Next, we are going to explain the decisions taken due to the results of the survey.

4.3. APPLICATION TO THE PROJECT

Due to the survey previously done, some decisions taken so as to implement the application have changed. Here we are going to explain all of them.

First of all, one of the predetermined options of choosing a discount for the shops will be the discount type percentage.

Second, when a shop wants to select the quantity of the discount offered, the predetermined options, by the moment, will be between 0-10, as the shops have expressed in the survey.

Another decision is that when the shop introduces the quantity that the customer has spent in the purchase, there will be no minimum quantity so as to generate a new voucher to the customer. In the future, as there are shops that wants a minimum quantity, it could be implemented and fix a minimum quantity depending on the sector type of the shop. For example, if the sector type is jewelry, there will be a minimum quantity of 40€ so as to generate a voucher. By the moment, there is no minimum quantity.

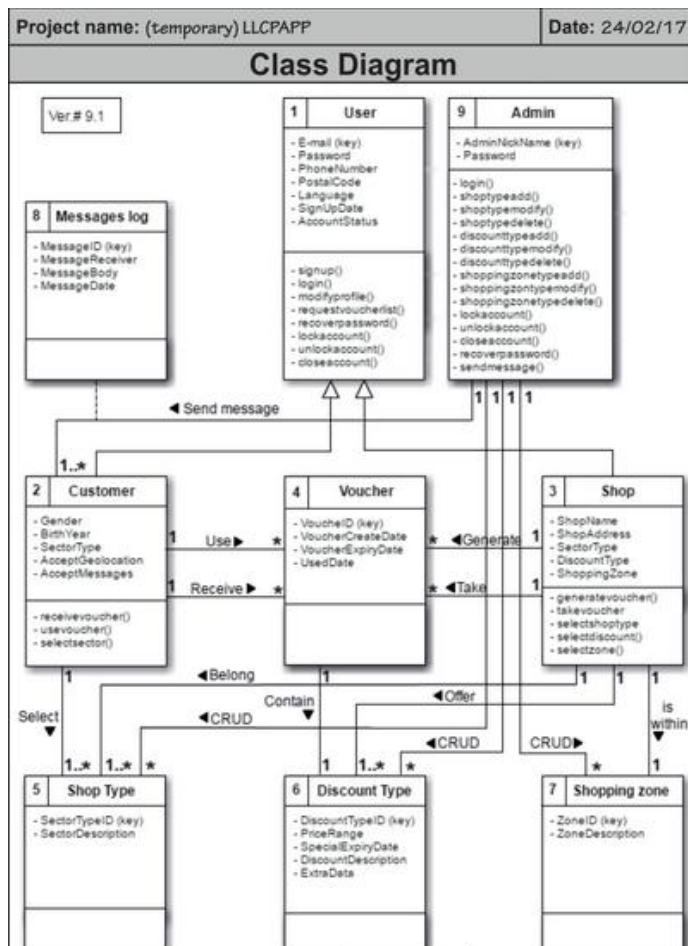
5. IMPLEMENTATION

5.1. THE DATABASE

5.1.1. DESIGN OF THE DATABASE

Before starting, I would like to say that all the knowledge used in the next explanation have been acquired during all the degree, especially in the subjects of database.

Before we started implementing the web application, we designed the database. This database is based on the class diagram from the project before. We have made some changes and updates, so as to improve the database, and make it more comfortable to implement. Next, we will explain the changes we have made. Here we have the class diagram in which we based.



First of all, we have changed the class name “Shop Type” to “Sector Type”, as we think it is a more representative name for the class.

Second, we have added some fields in some different classes, because it was necessary for the implementation. Here we explain in detail the modifications:

- In the classes `discountType`, `shopType` and `shoppingZone` we have added the field `name`, so as to have a representation of the class.
- In the class `Voucher` we have added the foreign keys to make reference to the classes that have a reference in it.
- In the class `Customer`, we have deleted the field `sectorType`, because the relation between classes is many to many, and we don't need a reference, as we create a register to another class every time a sector type is added to a customer.

Finally, we have added some new classes, so as to store some data needed for the implementation. These changes will be explained in more detail below.

Next, we can see the final database.

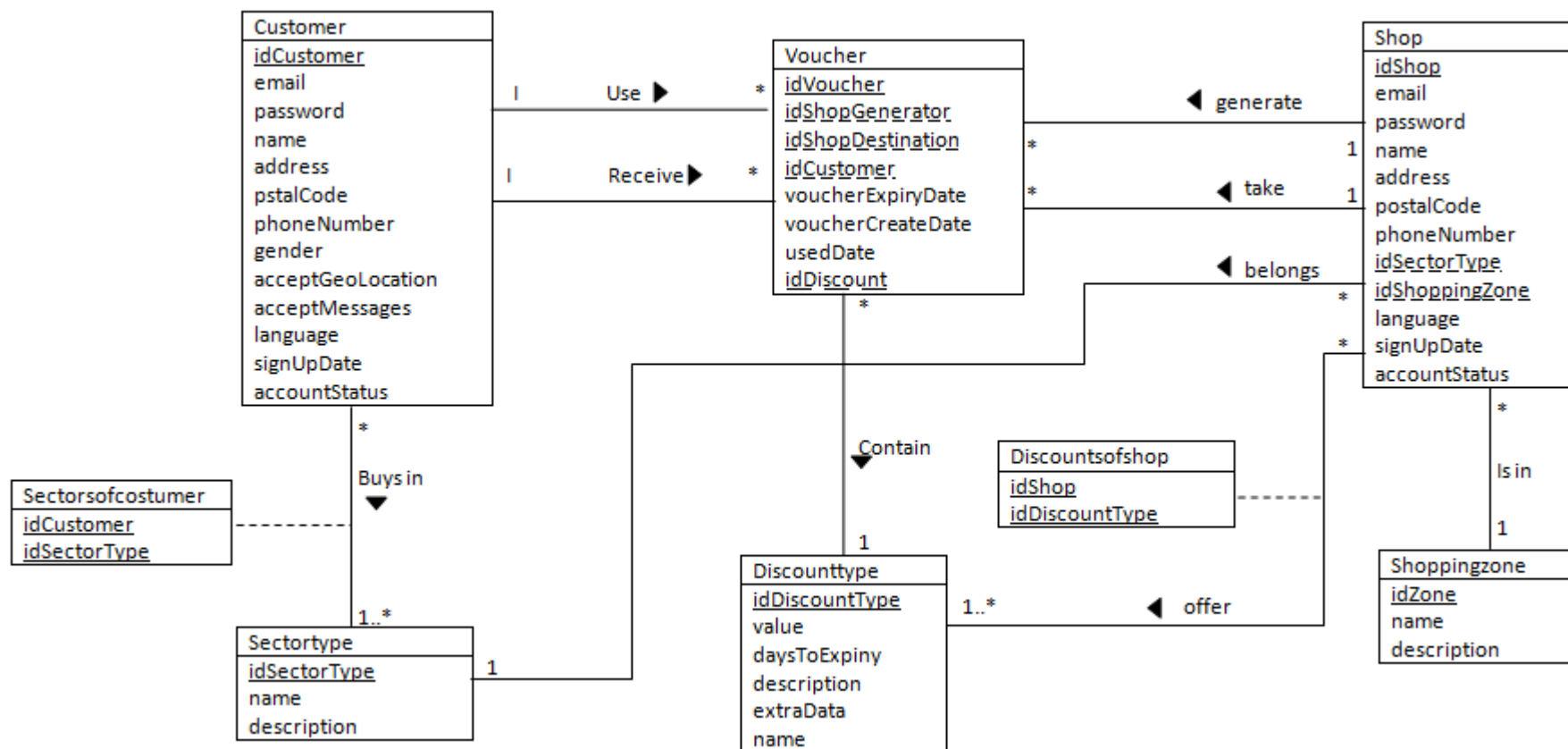


Table 1 Representation of the database

Here we can see the different classes and all their fields in detail:

Customer(idCustomer, email, password, name, address, postalCode, phoneNumber, gender, acceptGeoLocation, acceptMessages, language, signUpDate, accountStatus)

SectorType(idSectorType, description, name)

Sectorsofcustomer(idCustomer, idSectorType)

Shoppingzone(idZone, description, name)

Discounttype(idDiscountType, value, daysToExpiry, description extraData, name)

Shop(idShop, email, password, name, address, postalCode, phoneNumber, idSectorType, idShoppingZone, language, signUpDate, accountStatus)

Discountsofshop(idShop, idDiscountType)

Voucher(idVoucher, idShopGenerator, idShopDestination, idCustomer, voucherExpiryDate, voucherCreateDate, usedDate, idDiscount)

5.1.2. CLASSES

CUSTOMER

The customer is one of the possible users of the application. He will be able to see and receive the vouchers. The primary key is the field *idCustomer*, which is unique and can't be null.

SECTOR TYPE

This class represents the different sectors that exist nowadays in the market. Each shop will be in a sector type and every customer can buy in different sector types. The primary key is *idSectorType*, and a difference between the class diagram is that we have added a field called *name* so as to make the class more representative.

This class, as we explained before, has a relationship with the class customer and also with the class shop. This relationship with

SECTORS OF CUSTOMER

A register of this class is created every time that a customer adds a new sector type. This class is created so as to store different registers of every customer, because every customer can buy in more than one sector type. This class is born due to the relation between *customer* and *sectorType*, and its multiplicity is of 1 to many.

As a result of the relation, this class has a composed primary key. It is composed by the fields *idCustomer* and *idSectorType*. These fields are also foreign key of *customer* and *sectorType*

SHOPPING ZONE

This class describes the zone in which the shop is located. The primary key is *idZone*, and a difference between the class diagram is that we have added a field called *name* so as to make the class more representative.

DISCOUNT TYPE

This class describes the different possibilities of discounts. The primary key is *idDiscountType*. We have changed the field *priceRange*, for the field *value*. The field *value* indicates the amount of the discount. For instance, if the discount type is a percentage and the value is 5, it will represent a discount of the 5%. This modification has made the implementation easier. Besides, we have also changed the field *specialExpiryDate* for the field *daysToExpiry*. This field indicates how much time this discount type lasts. For instance, if this field value is 30, this discount will last for 30 days since the day of the creation. We have also added a field called *name* so as to make the class more representative.

SHOP

The shop is the second possible user of the application. A shop will be able to generate and post the vouchers. The primary key is *idShop*. The field *idSectorType* is a foreign key of the class *SectorType*. We store a reference to the class *SectorType* so as to know in which sector the shop belongs. The field *idShoppingZone* is a foreign key of the class *shoppingZone*. We store a reference to the class *shoppingZone* so as to know in which zone the shop is located.

DISCOUNTS OF SHOP

A register of this class is created every time that a shop adds a new discount type. This class is created so as to store different registers of every shop, because every shop can offer more than one discount type. This class is born due to the relation between *shop* and *discountType*, and its multiplicity is of 1 to many.

As a result of the relation, this class has a composed primary key. It is composed by the fields *idShop* and *idDiscountType*. These fields are also foreign key of *shop* and *discountType*.

VOUCHER

This class represents a voucher. It will contain all the data of every voucher. The primary key is *idVoucher*. The field *idShopGenerator* is a foreign key of the class *shop* and indicates the shop that has generated the voucher. We store a reference to the class *shop*, so as to know which shop has generated this voucher. The field *idShopDestination*, is a foreign key of the class *shop* and indicates which shop will receive the voucher, and the shop that the customer will use the voucher. We store a reference to the class *shop* so as to know which shop will receive the voucher. The field *idCustomer*, is a foreign key of the class *shop* and indicates the customer that receives the voucher. The field *idDiscount* is a foreign key of the class *idDiscountType* and indicates the type of the discount that the voucher is. We store a reference to the class *idDiscountType* so as to know more details of the voucher that has been created. Besides, we have the field *usedDate*, so as to know whether the voucher has already been used or not. If the value of the field is “NULL” it will mean that the voucher has not been used. Otherwise, if the field has a date, it will mean that it has already been used.

5.1.3. RELATIONSHIPS

Next, we are going to explain the different relationships between the classes that we have created so as to store all the data needed to display in the application.

CUSTOMER-VOUCHER

This relationship is between the class *customer* and the class *voucher*. A customer can either receive vouchers or use them. A customer can use or receive 0 or more than one vouchers. However, a voucher can only be assigned to one customer. So here we have a one-to-many relationship.¹

¹ In systems analysis, a one-to-many relationship is a type of cardinality that refers to the relationship between two entities A and B in which an element of A may be linked to many elements of B, but a member of B is linked to only one element of A.

So, the class voucher has a reference (foreign key) to the class customer. This foreign key is *idCustomer*, so every time that a voucher is created, we have to store the id of the customer that this voucher is given to.

CUSTOMER-SECTORTYPE

This relationship is between the class customer and the class sectorType. A customer can buy in 1 or more than one sector types. For instance, a customer can buy in the sector of jewelry and the sector of clothes. Besides, a sector type can have more than one customer, so here we have a relation of many-to-many.²

So, we have created a junction table called *sectorsofcustomer* in which we store a register every time that a customer adds a sector type. This register contains the id of the customer and the id of the sector type. With this, we will be able to know all the customers that a sector type has, and all the sectors in which a customer buys.

SHOP-SECTORTYPE

This relationship is between the class shop and the class sectorType. A shop can belong to only one sector type. However, a sector type can have more than one shop. This is a relationship of many-to-one. This kind of relationship is the opposite of a one-to-many relationship.

So, we have created the field *idSectorType* in the class shop, a reference to the class sectorType. This field is a foreign key.

SHOP-VOUCHER

This relationship is between the class shop and the class voucher. A shop can generate and take 0 or more than one vouchers. However, a voucher can only be assigned to one shop. This is a relationship of one-to-many.

So, we have created the fields in the class voucher called *idShopGenerator* and *idShopDestination*. These two fields are a reference to the class shop. They are foreign

² In systems analysis, a many-to-many relationship is a type of cardinality that refers to the relationship between two entities A and B in which A may contain a parent instance for which there are many children in B and vice versa.

In a relational database management system, such relationships are usually implemented by means of an associative table (also known as junction table or *cross-reference table*), say, AB with two one-to-many relationships A → AB and B → AB. In this case the logical primary key for AB is formed from the two foreign keys.

keys. We have created two references, because we need to store the data of the shop that is creating the voucher and the shop destination of the voucher.

SHOP-SHOPPINGZONE

This relationship is between the class shop and the class shoppinZone. A shop can be in one shopping zone, and a shopping zone can have zero or more than one shops. This is a relationship of many-to-one.

So, we have created a field in the class shop called *idShoppingZone*. This field is a reference to the class ShoppingZone. It is a foreign key.

SHOP-DISCOUNTTYPE

This relationship is between the class shop and the class DiscountType. A shop can offer 1 or more than one types of discounts and a discount type can be offered in one or more than one shops. This is a many-to-many relationship.

So, we have created a junction table called *discountsOfShop* in which we store a register every time that a shop adds a discount type. This register contains the id of the shop and the id of the discount type. With this, we will be able to know all discounts that a shop offers, and all the shops that are offering the discount.

VOUCHER-DISCOUNTTYPE

This relationship is between the class voucher and the class discountType. A voucher can contain only one discountType, and a discountType can be in 0 or more than one vouchers. This is a relationship of many-to-one.

So, we have created a field in the class voucher called *idDiscount*. This field is a reference to the class discountType. It is a foreign key.

5.2. IMPLEMENTATION OF THE DATABASE

So as to implement the database, we have used MySQL. The MySQL software delivers a very fast, multi-threaded, multi-user, and robust SQL (Structured Query Language) database server. MySQL Server is intended for mission-critical, heavy-load production systems as well as for embedding into mass-deployed software. ("MySQL :: MySQL 8.0 Reference Manual :: 1 General Information", 2018)

So, the database is implemented in MySQL. So as to handle the administration of the database, we have used the tool phpmyadmin³.

³ phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement.

6. THE SERVER

So as to implement the server, we have used PHP.

Hypertext Preprocessor (or simply PHP) is not only a server-side scripting language designed for web development but also used as a general-purpose programming language. It was originally created by Rasmus Lerdorf in 1994. The PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Preprocessor. PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems, and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications. ("PHP: History of PHP and Related Projects - Manual", 2018)

Personally, I had never used this language during these 5 years of the double degree. The reason why I decided to implement it this way is because, after doing a research on the way to implement a server, I found that PHP was quite easy to learn, and as I had never used it before, I found quite interesting to learn a new language and use it for the first time. So, I started learning php with different video tutorials and some manuals. After that, and some tests, I could start implementing the server.

In computing, a server is a computer program or a device that provides functionality for other programs or devices, called "clients". This architecture is called the client-server model, and a single overall computation is distributed across multiple processes or devices. Servers can provide various functionalities, often called "services", such as sharing data or resources among multiple clients, or performing computation for a client. A single server can serve multiple clients, and a single client can use multiple servers. A client process may run on the same device or may connect over a network to a server on a different device.[1] Typical servers are database servers, file servers,

mail servers, print servers, web servers, game servers, and application servers. (Comer & Stevens, 1993)

Next, I'm going to explain the different implementation decisions that have been taken, so as to implement the server. To make a more comfortable implementation of the server, the server has been divided in a server for the shops, and a server for the customer. This is a database server, and it is used so as to store all the data of the users in the database.

6.1. REGISTER

In this part, we store all the data received in the register page. Before the user starts using the application, he has to register in the web. In this register he has to introduce different personal information that can be useful in the application. We store this data with some access to the database, using MySQL, as we explained before. We have implemented all kind of possible errors that the register can have. For instance, if a required field is not filled, an error message will be displayed. Moreover, the password must be written twice. If this password does not match, a message error is displayed. Besides, if the email is already registered in the database, an error message will also be displayed.

6.2. SELECT DISCOUNT TYPE

This part was pretended to be selected during the register of a shop. We have made a change, and we have created another step during the register, so as to select the discount type. After doing the market analysis, we wanted to focus more on the discounts, and let the shops select the discounts apart from the register page. In this part, we store the information in the database that the shop selects. All kind of errors have been implemented. For instance, if one of the required fields is not filled, an error message is displayed.

6.3. LOGIN

This part is used when a user wants to access in the web application. It checks whether the user and password are valid or not. In case the password or the email is not valid, an error message is displayed. In case the user has been locked, an error message is

displayed. This message tells the user that he has to contact with an administrator. Besides, if a required field is not filled, an error message is also displayed.

6.4. MODIFY ACCOUNT

This part is used when a user wants to modify the information of his account. First of all, the current password is needed so as to make any kind of change. We have made another change in this part. We have removed the part of selecting the discount type or adding the sector type and we have created another page, so as to make the implementation easier and clearer. These modifications are made with an access to the database and updating the old values with the new values of the fields that have been changed. The current password field is required, so if it is not filled, an error message is displayed.

6.5. ADD DISCOUNT TYPE

This part has been added instead of doing it in the part of modify account. In this part, the user is able to add a new discount type. We store the data information to the database. We check all possible errors. For instance, if a field required is not filled, an error message is displayed.

6.6. CLOSE ACCOUNT

This part closes an account. It deletes the data from the database, and then the user will be completely erased. There is a required field that is the password. If this field is not filled in, an error message is displayed.

6.7. LOCK ACCOUNT

This part is used to lock an account. It sets the field *accountStatus* of a user to the value "LOCKED". The account will be permanently unavailable, unless the owner of the account contacts an administrator so as to unlock it. There is a required field that is the password. If this field is not filled in, an error message is displayed.

6.8. TAKE VOUCHER

This is one of the most important parts of the functionality of the application. It is used to take a voucher from a customer. This part receives the code of a voucher and the quantity that the customer has spent. First of all, it checks whether the voucher exists

in the database. It also checks if the voucher can still be used, checking the field *expiryDate* and if this field is "NULL" it means that the voucher has not expired and can still be used. It also checks if the shop destination of the voucher is the same shop as the one taking the voucher. If one of these possible situations happens, an error message is displayed. Moreover, if everything is correct, the field of the voucher is updated and the field *usedDate* is set to the current date.

Besides, another functionality is done in this part. When a shop takes a voucher from a customer, a new voucher is automatically generated. This part when the customer loyalty is increased and the customers are encouraged to buy in other shops in Lleida. When a customer makes a purchase of a product, he automatically receives a voucher to another shop. We have created an algorithm so as to auto-generate a new voucher. This function is called *giveVoucherFunc*, and it is explained below. Finally, if everything is correct, a success message is displayed.

6.9. GIVE VOUCHER

This is the other important part of the server. Its functionality is to give a voucher to a customer when it is the first time he buys in this shop, and does not have a voucher to use. It receives the email of the customer and the quantity that the customer has spent in the purchase. First of all, it checks whether the email of the customer really exists and that the amount of money is valid. Then, it creates a new voucher to the customer, with the function *giveVoucherFunc* that we explain below. Finally, if everything is correct, we display a success message.

6.9.1. GENERATING A VOUCHER

It is an important part, to know how a voucher is generated. To generate a voucher, we have had some different conditions.

The first one is that when a shop generates a new voucher, the sector type of this shop can't be the same sector type as the shop that receives the voucher. For instance, imagine that the shop A belongs to the sector of jewelry and it generates a new voucher. This voucher is generated to the shop B, but the sector type of the shop B can't be the sector of jewelry. It could be, for example, to the sector of bars. So, before creating the voucher, we have to assure that the new shop chosen does not belong to the same sector. The reason why we have taken this decision is because the

application does not want to create competence between shops of the same sector type. From the point of view of the shops, we think that they wouldn't use an application that creates a discount to their directly competence, just because a customer has made a buy in your own shop. Then, we have decided that the shops will generate new vouchers, but only to the shops of a different sector.

The second condition is that the voucher created will depend on how much money the customer has spent in the shop. The more money a customer spent, the bigger will be the amount of the discount received. For instance, if a shop offers discounts of 5% and 20%, if a customer spends a huge quantity of money, he will receive the discount of the 20%. Otherwise, if he spends few money, he will receive the discount of the 5%.

Next, we explain how we generate a voucher and how we have implemented these two conditions in the code, so as to make them possible. We have created different functions:

giveVoucherFunc:

This function has been created so as to give a voucher to a customer, given the current shop, the customer email and the quantity of the buy. First of all, we keep the data required so as to create a register in the database of a voucher. This is the *id* of the shop generator, the *id* of the shop destination, the *id* of the customer the expiry date, the current date and the *id* of the discount type. Then, when we have all this information, we create a new voucher, generated by the current shop, to the customer that has made a buy to the shop. This function is also used when a customer uses a voucher in a shop. Then, automatically, a new voucher is generated to another shop. Next, we are going to explain how we find the shop that will be the destination of the voucher.

getShopByPreferencesCustomer:

First of all, we select the different sector types of the customer, by accessing the database. This sector types must be different from the current sector type of the shop. In case the customer has only the same sector type, no vouchers can be generated. Once we have selected all the sector types of the customer, we choose one of them randomly.

Next, we have to select one shop that belongs to this sector type. When we have all the shops belonging to this sector type, we select one of them randomly. Finally, we get the id of the shop destination of the voucher.

Next, we are going to explain how we find the discount type of the new voucher that we are generating.

getDiscountTypeByShop:

First of all, we select the different discount types that the shop destination has. We store them in an array. We order them by the value of the discount from the smallest to the biggest. Then, we select the discount type, depending on the quantity that the customer has spent in his buy. If the quantity is less than 50€ we will select the discount from the first part of the array, so it will be the lowest one. If the quantity of the buy is bigger than 50€, we will select the discount from the second part of the array, so the biggest one.

6.10. SELECT SECTOR TYPE

This part was pretended to be selected during the register of a customer. We have made a change, and we have created another step during the register, so as to select the sector type. We have done this so as to make an easier implementation. In this part, we store the information in the database that the customer selects. All kind of errors are have been implemented. For instance, if one of the required fields is not filled, an error message is displayed.

6.11. ADD SECTOR TYPE

This part has been added instead of doing it in the part of modify account. In this part, the user is able to add a new sector type. We store the data information to the database. We check all possible errors. For instance, if a field required is not filled, an error message will be displayed.

7. MYSQL INJECTION

SQL injection refers to the act of someone inserting a MySQL statement to be run on your database without your knowledge. Injection usually occurs when you ask a user for input, like their name, and instead of a name they give you a MySQL statement that you will unknowingly run on your database. ("MySQL Tutorial - SQL Injection", 2018)

SQL injection is one of the most common web hacking techniques.

For example, an attacker could empty out a table by executing a DELETE statement.

This example below explains how:

Imagine we have a statement in our code like:

- `"SELECT * FROM customers WHERE username = $username`

In which we select the username that a user of the application has entered via input.

But now, imagine that instead of entering the username, the user enters a statement like this:

- `""; DELETE FROM customers WHERE 1 or username = ""`

This last statement will automatically finish the statement in our code, and will execute the statement of deleting all the table customers. With this statement, the table customers would be erased from the database.

To prevent this problem, we have used a command that escapes special characters in a string for use in an SQL statement. This command is:

- `string mysql_real_escape_string (string $unescaped_string [, resource $link_identifier = NULL])`

What `mysql_real_escape_string` does is take a string that is going to be used in a MySQL query and return the same string with all SQL Injection attempts safely escaped. Basically, it will replace those troublesome quotes(') a user might enter with a MySQL-safe substitute, an escaped quote `\'`. ("PHP: MySQL Real Escape String - Manual", 2018)

8. THE WEB APPLICATION

The whole project is thought to work in devices that the shops are able to have. These devices could be for example a tablet or a mobile phone. This second phase of the whole project has focused on implementing a prototype of the application. This prototype has been created via web application. The implementation of the application has been done in PHP and HTML. In this section, we are going to explain the different pages of the web application that have been implemented. These pages are based on the pages previously designed by Jorge. However, there are some changes and updates. These changes and updates have been done so as to improve the application and make an easier implementation. Before we start explaining the pages, we have to say that this web application is totally functional, so there is no web design behind it. This prototype is done before doing the design of the application and before implementing the real application. It is done to see how it would work and to check if the shops are interested in continuing the project.

Next, we are going to explain the most important pages of the application and the changes that have been done.

The application has a main page or home page in which we can select all the possible menus of the application. Previously, we will have had logged in.

SHOP HOME PAGE

As we said before, in the home page of the shop, we find all the possible menus that the application has. Here we can see a capture of the real web application. We can see the menu of take voucher and the menu of give voucher, which the shop will have to use so as to give a voucher to the customer.

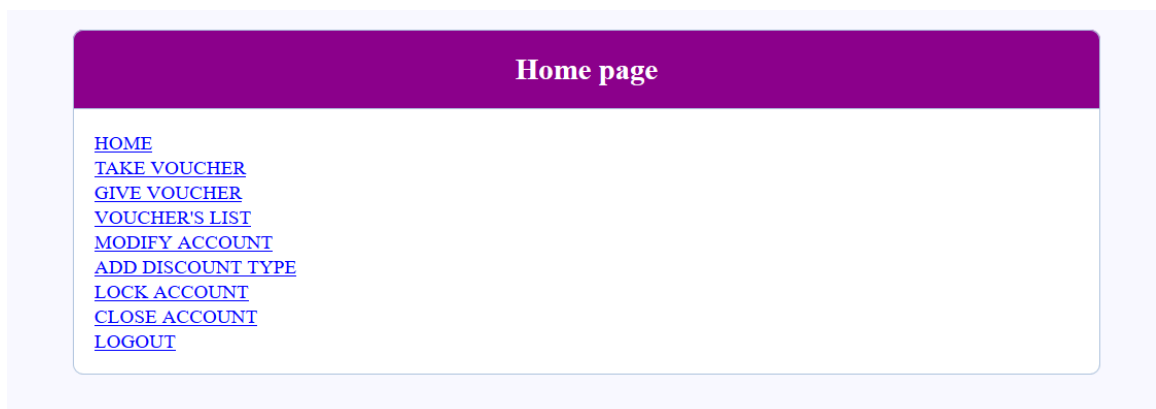
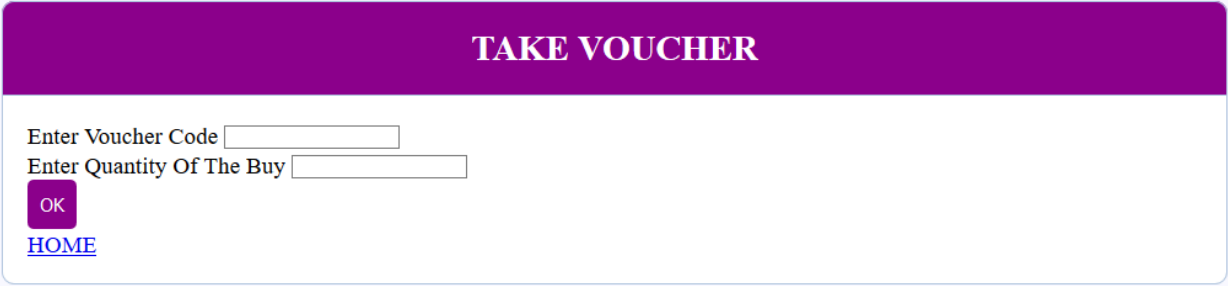


Image 1 Shop home page

SHOP TAKE VOUCHER

In this page we have done a little update to the previous prototype. In this page, we have added a new field. This field is the quantity that the customer has spent in his/her buy. With this field, we will be able to know how much money the customer has spent, and we will be able to give him one kind of voucher or other. For example, if the customer spends a big quantity of money, he will receive a bigger discount.



TAKE VOUCHER

Enter Voucher Code

Enter Quantity Of The Buy

[HOME](#)

Image 2 Shop Take Voucher

SHOP GIVE VOUCHER

In this case, we have also done a little update. We have also added the field quantity of the buy, so as to know how much money has the customer spent.

SHOP ADD DISCOUNT TYPE

This page was not created in the prototype before. As we were implementing the web application, we realized that making this page apart was better. This functionality was initially done in the modify account page, but as we want to emphasize more with the discounts, we have created this new page. Here we can see the new page:

Image 3 Shop Add discount type

In it, we can see the current discounts that the shop is offering. This page is done so as a shop can add a new discount type that he wants to offer. We can see that the discount type can be selected, and also the value of this discount.

CUSTOMER HOME PAGE

In this page, the customer will be able to see all the possible menus of the application. It is quite similar to the home page of the shop, but in this case, the customer has the menu use voucher and get voucher instead of the menus take voucher and give voucher.

Image 4 Customer home page

CUSTOMER USE VOUCHER

This page has been implemented as it was designed but some fields have been added. We can see that it shows the code of the voucher that the customer has to use. We

can also see the QR code. This QR code contains the code of the voucher. Theoretically, when customer has to use a voucher, he has to show this page, and the shop has to electronically capture the QR code, and automatically the voucher will be used and a new one will be generated. In case that the shop does not have a QR code reader, he can also introduce manually the voucher code shown in this page. We have also added two fields more. The first field is the description of the code, so as to know what kind of discount must be applied. The second field is the shop destination of the voucher, so as to assure that the voucher is for this shop. Here we can see a capture of the real web application.

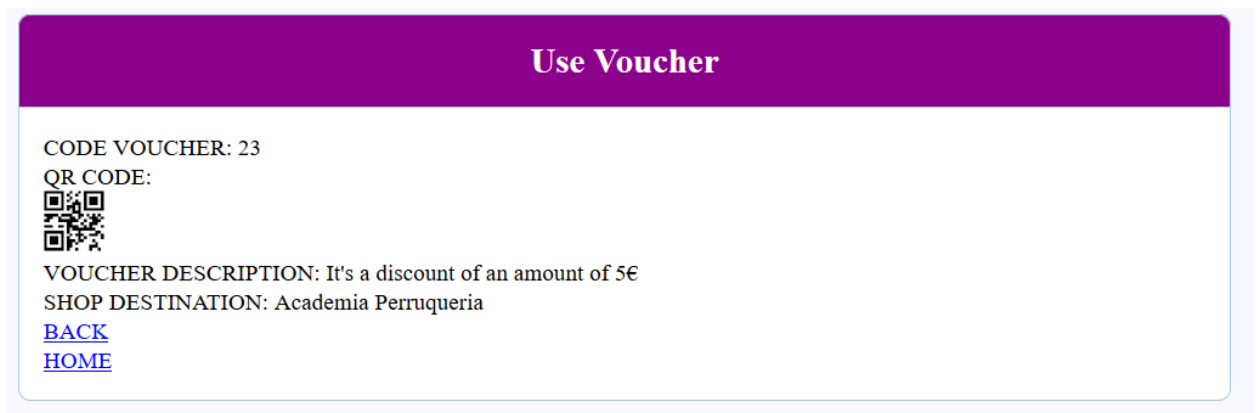


Image 5 Customer Use voucher

CUSTOMER GET VOUCHER

This page is similar to the one designed in the prototype but we have added some extra data. In this page, we can see a QR code. This QR code must be captured by the shop, so as to know the data of the customer. This QR code indicates the email of the customer and then when it is captured, a voucher is automatically generated to that customer. In case that the shop does not have a QR code reader, he can introduce the email of the customer manually, as it is also shown in this page. Here we can see the real page of the web application:

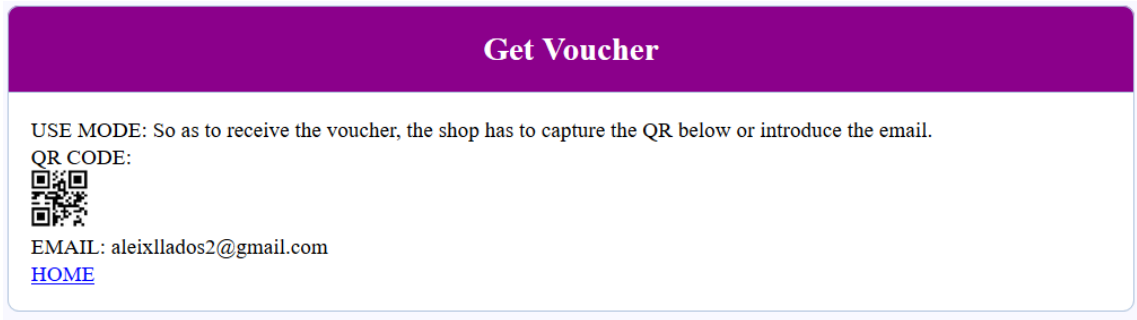


Image 6 Customer Get voucher

CUSTOMER ADD SECTOR TYPE

This page was not created in the prototype before. As we were implementing the web application, we realized that making this page apart was better. This functionality was initially done in the modify account page, but as we wanted to emphasize more with the sector types, we created this new page. Here we can see the new page:

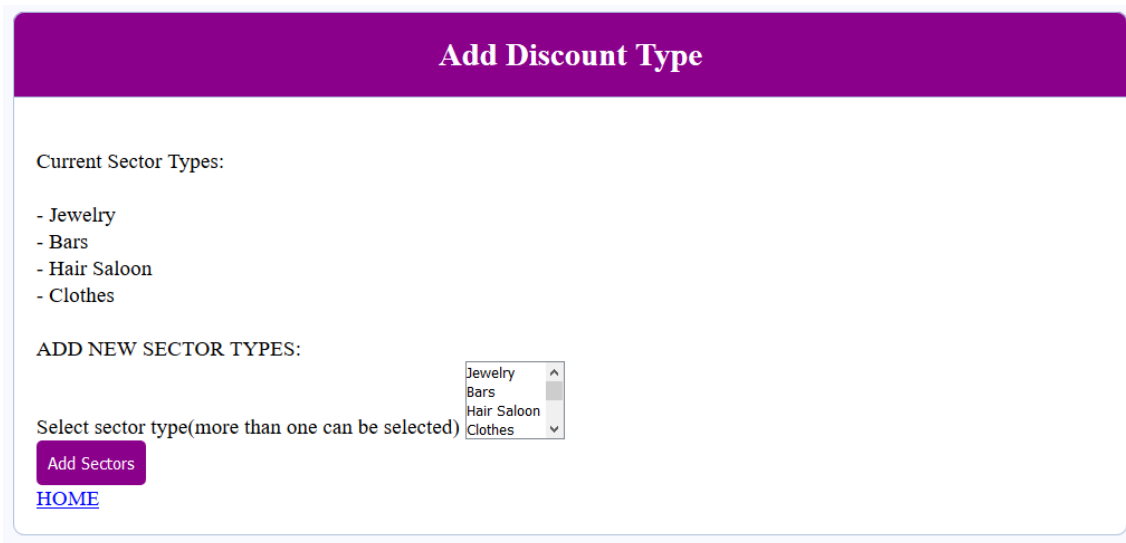


Image 7 Customer Add sector type

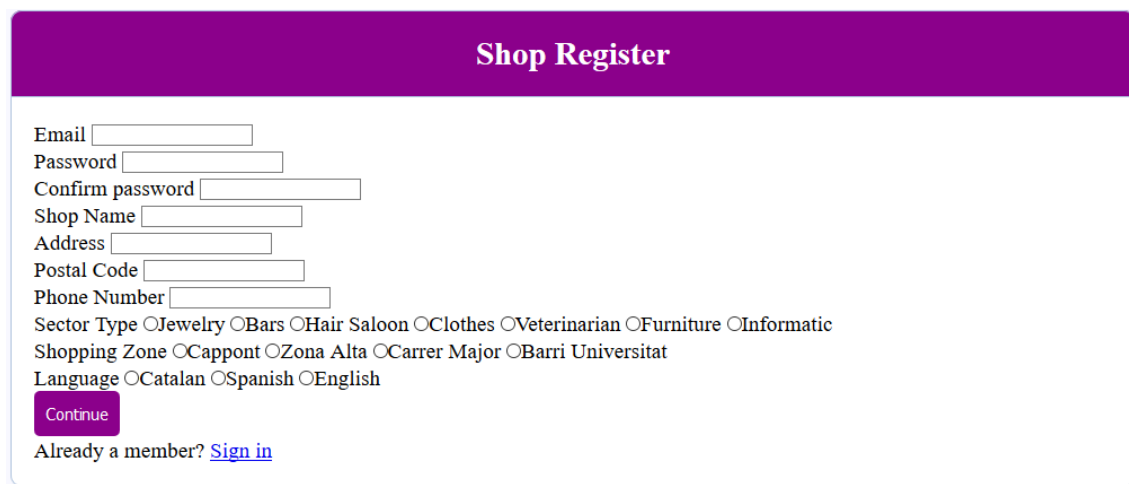
In it, we can see the current sector types in which the customer wants to receive discounts. In this page, a customer can add new sector types in which he wants to receive vouchers. We can see that more than one sector type will be selected.

The rest of the pages of the web application are trivial, and have been implemented exactly as they were designed in the previous project. Here we can see all of them:

SHOP LOGIN

The Shop Login form features a purple header with the text "Shop Login". Below the header, there are two input fields: "Email" and "Password". A purple "Login" button is positioned below the password field. At the bottom, there is a link "Not a member yet? [Sign up](#)".

Image 8 Shop Login

SHOP REGISTER

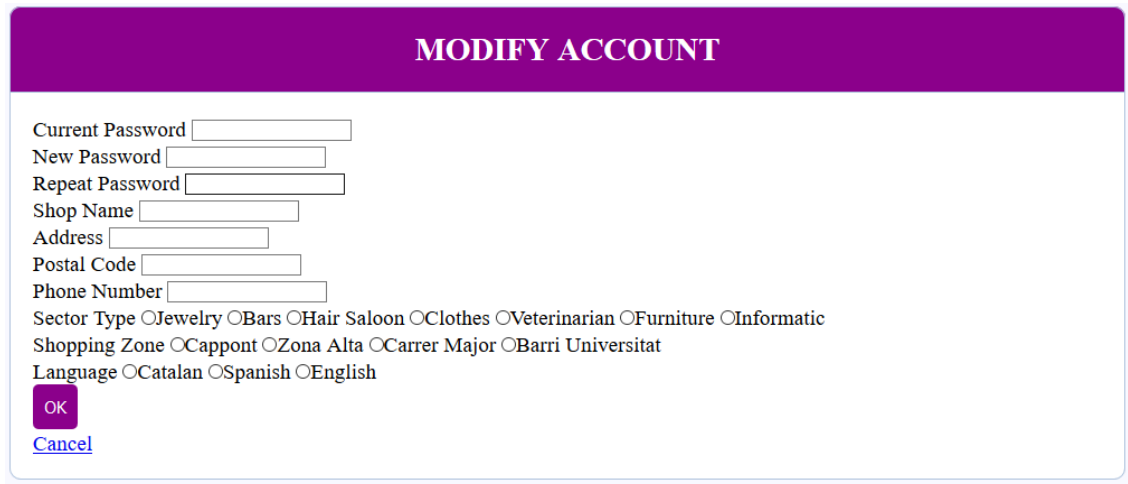
The Shop Register form has a purple header with the text "Shop Register". The form includes input fields for "Email", "Password", "Confirm password", "Shop Name", "Address", "Postal Code", and "Phone Number". Below these fields are several radio button options: "Sector Type" (Jewelry, Bars, Hair Saloon, Clothes, Veterinarian, Furniture, Informatic), "Shopping Zone" (Cappont, Zona Alta, Carrer Major, Barri Universitat), and "Language" (Catalan, Spanish, English). A purple "Continue" button is located at the bottom left. A link "Already a member? [Sign in](#)" is at the bottom right.

Image 9 Shop Register

SHOP VOUCHER'S LIST

The Shop Voucher's List form has a purple header with the text "Shop Voucher's List". Below the header, there are three links: "[VOUCHERS GENERATED](#)", "[VOUCHERS RECEIVED](#)", and "[HOME](#)".

Image 10 Shop Voucher's List

MODIFY SHOP ACCOUNT:

MODIFY ACCOUNT

Current Password

New Password

Repeat Password

Shop Name

Address

Postal Code

Phone Number

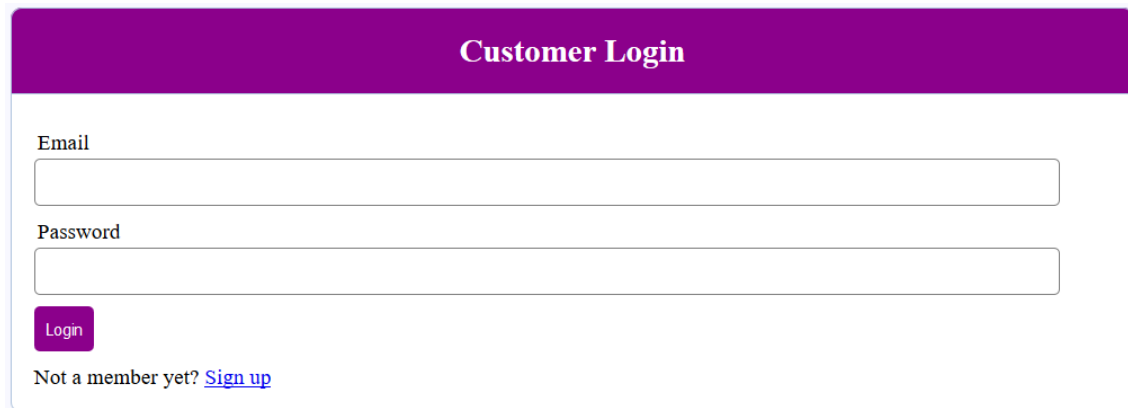
Sector Type ☐Jewelry ☐Bars ☐Hair Saloon ☐Clothes ☐Veterinarian ☐Furniture ☐Informatic

Shopping Zone ☐Cappont ☐Zona Alta ☐Carrer Major ☐Barri Universitat

Language ☐Catalan ☐Spanish ☐English

[Cancel](#)

Image 11 Modify shop account

CUSTOMER LOGIN:

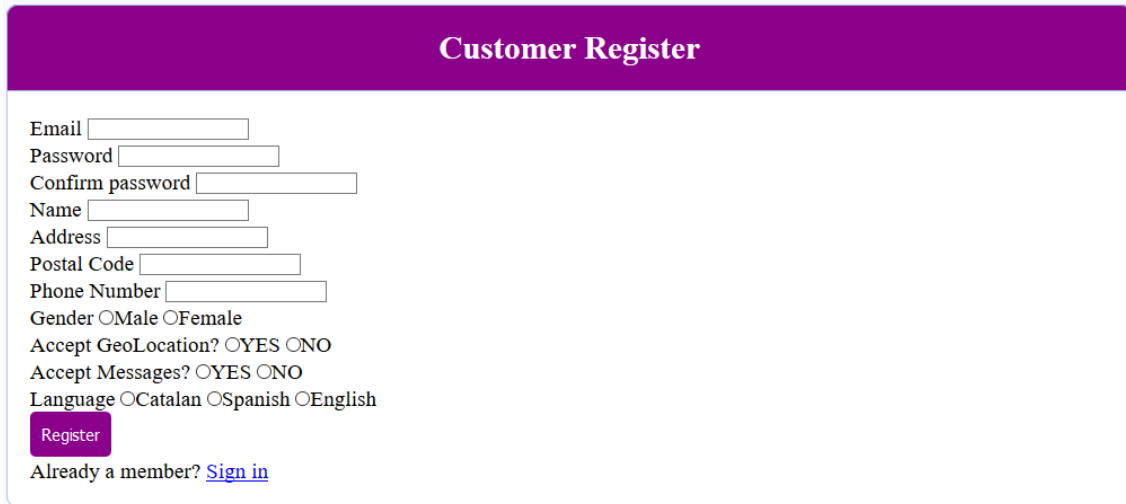
Customer Login

Email

Password

Not a member yet? [Sign up](#)

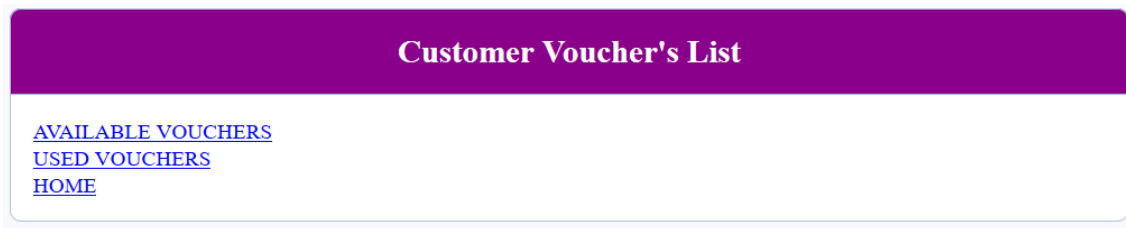
Image 12 Customer login

CUSTOMER REGISTER:

The screenshot shows a web form titled "Customer Register" with a purple header. The form contains the following fields and options:

- Email
- Password
- Confirm password
- Name
- Address
- Postal Code
- Phone Number
- Gender ☐ Male ☐ Female
- Accept GeoLocation? ☐ YES ☐ NO
- Accept Messages? ☐ YES ☐ NO
- Language ☐ Catalan ☐ Spanish ☐ English
-
- Already a member? [Sign in](#)

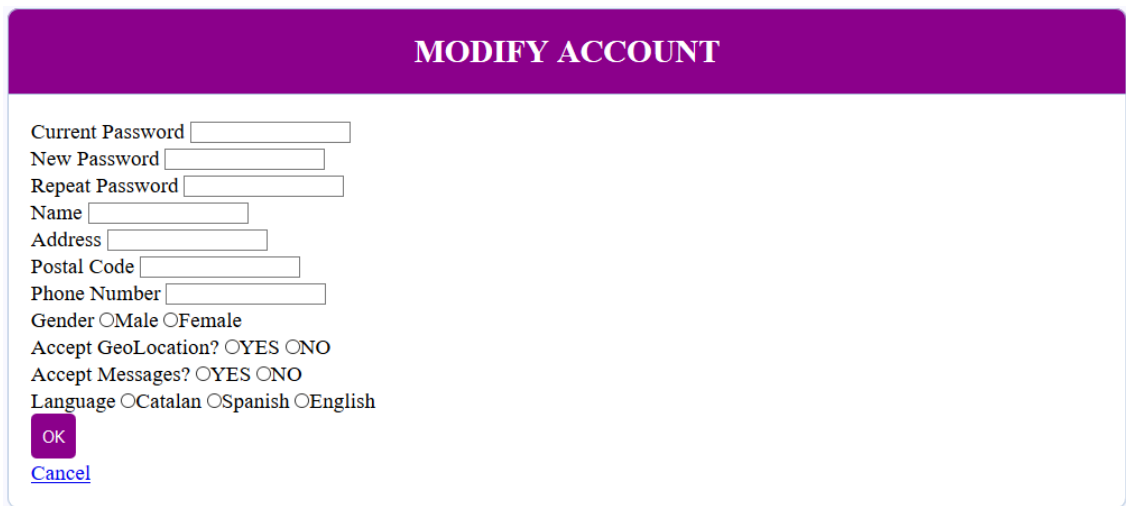
Image 13 Customer register

CUSTOMER VOUCHER'S LIST

The screenshot shows a web page titled "Customer Voucher's List" with a purple header. Below the header, there are three links:

- [AVAILABLE VOUCHERS](#)
- [USED VOUCHERS](#)
- [HOME](#)

Image 14 Customer Voucher's List

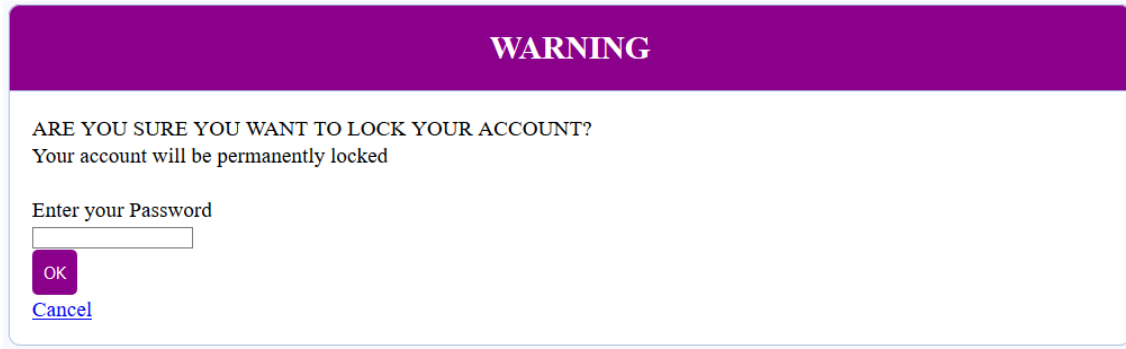
CUSTOMER MODIFY ACCOUNT

The screenshot shows a web form titled "MODIFY ACCOUNT" with a purple header. The form contains the following fields and options:

- Current Password
- New Password
- Repeat Password
- Name
- Address
- Postal Code
- Phone Number
- Gender ☐ Male ☐ Female
- Accept GeoLocation? ☐ YES ☐ NO
- Accept Messages? ☐ YES ☐ NO
- Language ☐ Catalan ☐ Spanish ☐ English
-
- [Cancel](#)

Image 15 Customer Modify Account

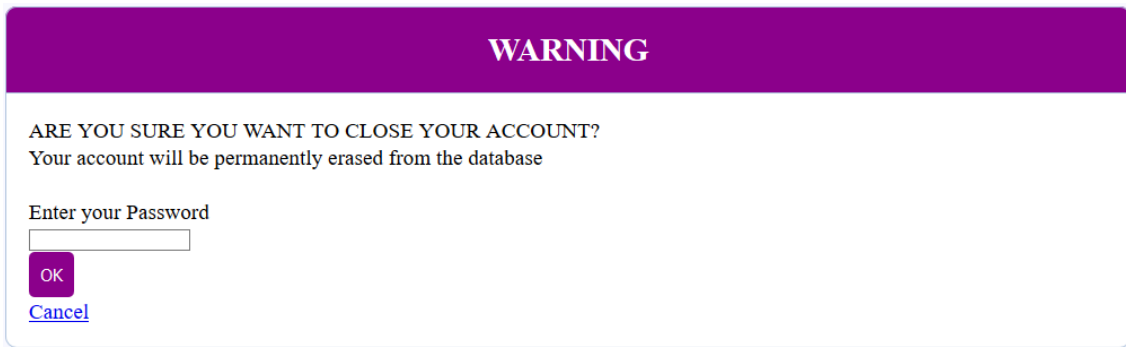
LOCK ACCOUNT



A warning dialog box with a purple header containing the word "WARNING". The main text asks "ARE YOU SURE YOU WANT TO LOCK YOUR ACCOUNT?" and states "Your account will be permanently locked". Below this is a label "Enter your Password" followed by a text input field. At the bottom are two buttons: a purple "OK" button and a blue "Cancel" link.

Image 16 Lock Account

CLOSE ACCOUNT



A warning dialog box with a purple header containing the word "WARNING". The main text asks "ARE YOU SURE YOU WANT TO CLOSE YOUR ACCOUNT?" and states "Your account will be permanently erased from the database". Below this is a label "Enter your Password" followed by a text input field. At the bottom are two buttons: a purple "OK" button and a blue "Cancel" link.

Image 17 Close Account

8.1. TESTING THE WEB APPLICATION

During the project, we have been testing all the functionalities of the application. Application testing is a process by which application software developed for handheld mobile devices is tested for its functionality, usability and consistency. Mobile application testing can be an automated or manual type of testing. As we went implementing the application, we were testing the functionality of the current page that was being implemented. We tested all kind of possibilities, so as to find all possible errors. By the moment, all the tests were for the functionality, because the application is only a prototype and it is only functional. We have tested the following:

Checking all the links:

- Tested the outgoing links from all the pages to specific domain under test.
- Tested all internal links.

- Tested links jumping on the same pages.
- Tested to check if there are any orphan pages.

Testing forms on all pages:

- First, checked all the validations on each field.
- Checked for default values of the fields.
- Wrong inputs in the forms to the fields in the forms.

Database testing:

- Data consistency is also very important in a web application. Checked for data integrity and errors while editing, deleting or modifying the forms.
- Checked if all the database queries are executing correctly, data is retrieved and also updated correctly.

So as to make these tests, we have created some fixed values of the different classes in the database, these values are the following:

- 3 customers with different sector type each of them.
- 3 shops with different sector types and offering different discount types.
- 10 different kinds of discount types, such as percentage, amount and discount for multiple products. Each of this discounts have different values.
- 7 different kind of sector types, such as Jewerly, Bars, Hair Saloon, Clothes, Veterinarian, Furniture or Informatic
- 4 different kinds of zones, such as Cappont, Zona Alta, Carrer Major, Barri universitat. Every of this register refers to a zone of lleida.

With all these predetermined values, we were able to execute all different kind of tests. Doing tests as we were implementing the web application has been very useful. We realized that there were functionalities not implemented yet, as the application went down. Maybe, this process has been one of the most difficult parts of the whole project. In my opinion, it is quite difficult to empathize with an user and think about all the possibilities that he can do. For example, what happens if a user enters a not valid code of voucher? What happens if a user enters a not valid quantity of the buy? These scenarios must be contemplated, and the only way to do it, is to imagine all the possible moves that the user can do. Every time that we figured out a possible

functionality of the application, we tried to fix it. We have created some additional pages, or pop up messages, in which we display information to the user to let him know that he is doing something wrong or right.

Next, we are going to explain and show all the different scenarios that can be found on the application:

TAKE/GIVE VOUCHER

Imagine a user only has registered one kind of sector type and he makes a purchase in a shop of that sector type. A voucher won't be created, because shops can't create vouchers to a shop of the same sector type, as it would create competence. Then, this user won't receive any new voucher. This message is displayed on the web application when this scenario happens. Here we can see message:

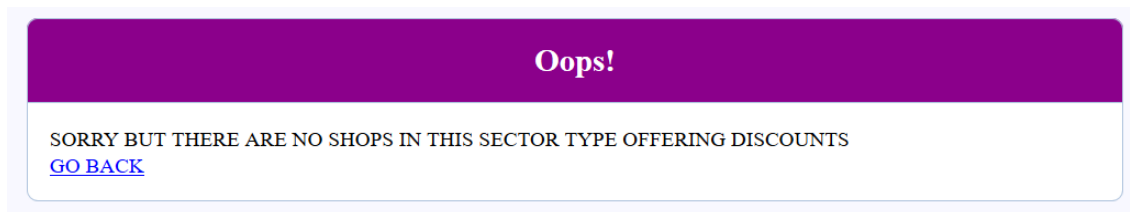


Image 18 Take/Give Voucher

This message is shown after the give voucher page.

In case everything is correct and the user receives a voucher, the page shown is the following:

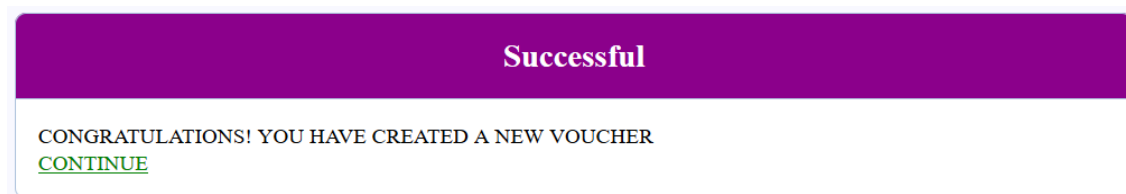
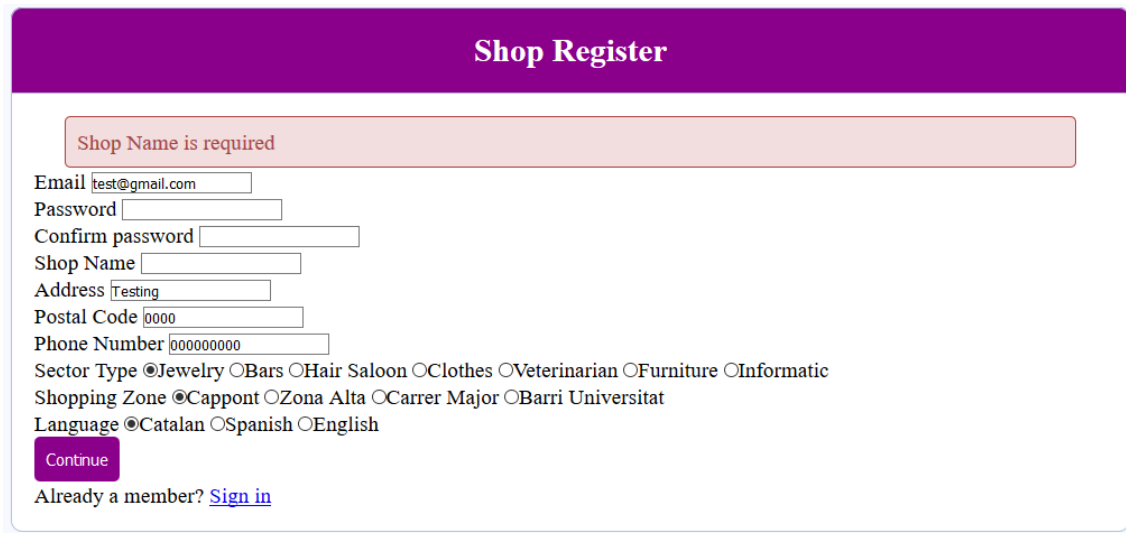


Image 19 Take/Give Voucher Successful

REGISTER USER

In the case of registering a user, no matter if it is a shop or a customer; there are some possible scenarios, which are all contemplated. During the process of registering, if any of the required fields is not filled, a message will appear in the register pages. Here we have a capture of the real web application:



The screenshot shows a web form titled "Shop Register" with a purple header. A red error message "Shop Name is required" is displayed at the top. The form contains the following fields and options:

- Email:
- Password:
- Confirm password:
- Shop Name:
- Address:
- Postal Code:
- Phone Number:
- Sector Type: ☒Jewelry ☐Bars ☐Hair Saloon ☐Clothes ☐Veterinarian ☐Furniture ☐Informatic
- Shopping Zone: ☒Cappont ☐Zona Alta ☐Carrer Major ☐Barri Universitat
- Language: ☒Catalan ☐Spanish ☐English

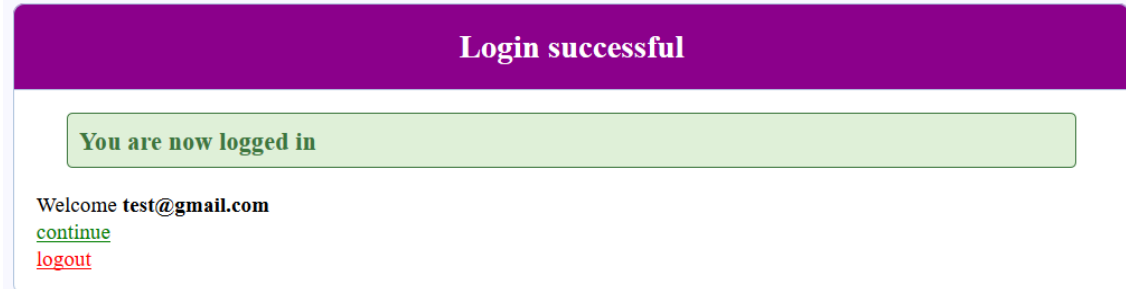
At the bottom, there is a purple "Continue" button and a link "Already a member? [Sign in](#)".

Image 20 Register User

In this case, the user has forgotten to write the name of the shop.

Another scenario is that the email entered by the user already exists. In this case, a message is displayed saying that the email already exists.

If everything is correct, a success message is shown before the user is redirected to the main page of the application. This success page is the following:



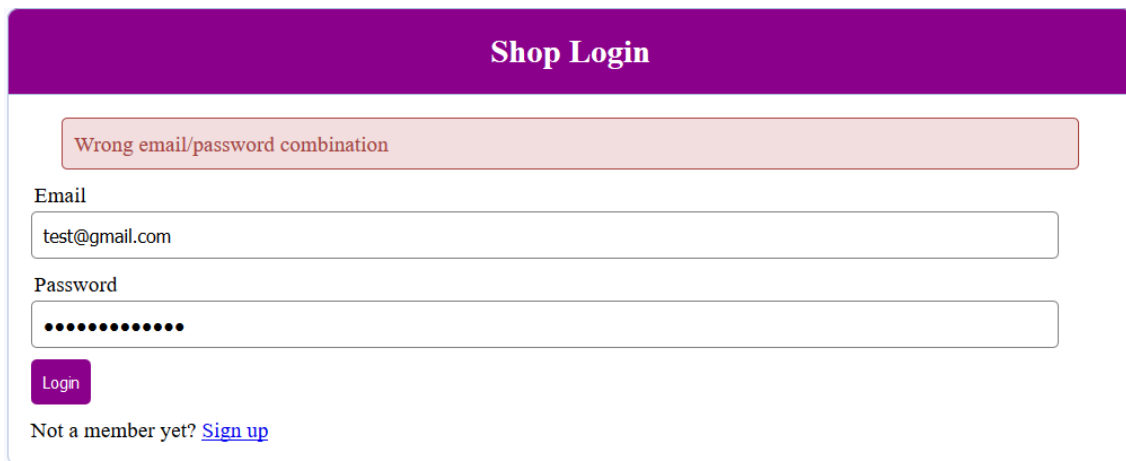
The screenshot shows a web page titled "Login successful" with a purple header. A green success message "You are now logged in" is displayed. Below the message, the text "Welcome test@gmail.com" is shown, followed by two links: "continue" (green) and "logout" (red).

Image 21 Login Successful

USER LOGIN

In the case of logging in, there are three possible scenarios.

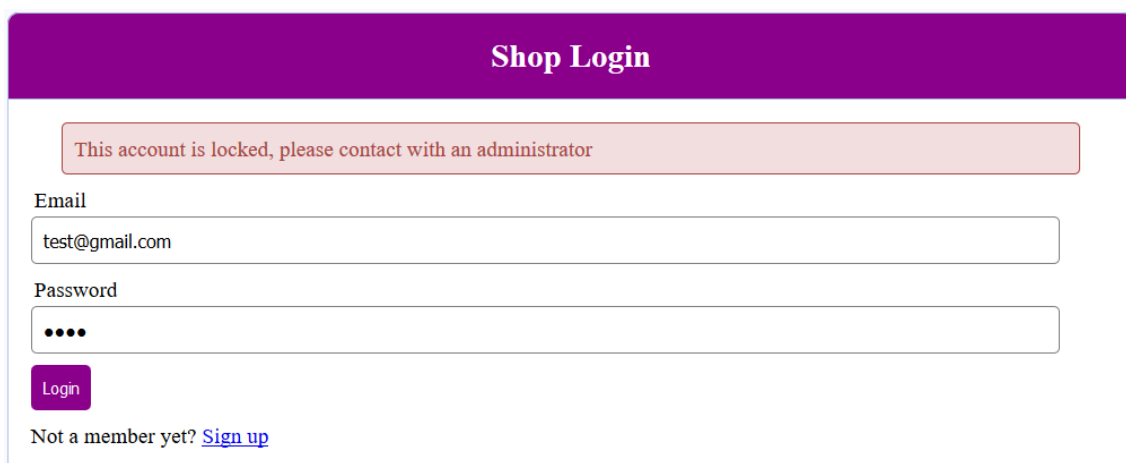
The first one is when the user enters a valid email or the password does not match with the email. In this case, a message is shown, saying that there is a wrong email/password combination. Here we can see the example:



The screenshot shows a web form titled "Shop Login" with a purple header. Below the header is a red error message box that says "Wrong email/password combination". The form contains two input fields: "Email" with the value "test@gmail.com" and "Password" with masked characters ".....". A purple "Login" button is positioned below the password field. At the bottom of the form, there is a link that says "Not a member yet? [Sign up](#)".

Image 22 User Login

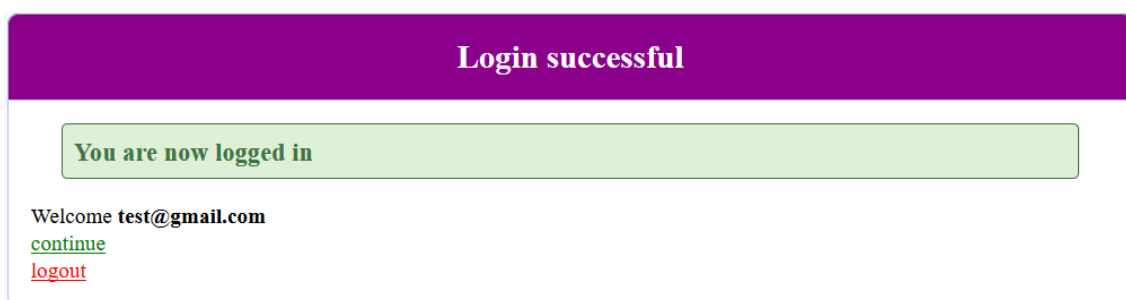
The second scenario is that the user has the account locked. This happens when the user has locked it previously. A message is shown, saying that the account is locked and the user has to contact with an administrator. Here we can see the example:



The screenshot shows a web form titled "Shop Login" with a purple header. Below the header is a red error message box that says "This account is locked, please contact with an administrator". The form contains two input fields: "Email" with the value "test@gmail.com" and "Password" with masked characters "....". A purple "Login" button is positioned below the password field. At the bottom of the form, there is a link that says "Not a member yet? [Sign up](#)".

Image 23 Shop Login

The third possible scenario is when everything is correct and the user can access into the application. Then, a successful page is shown:

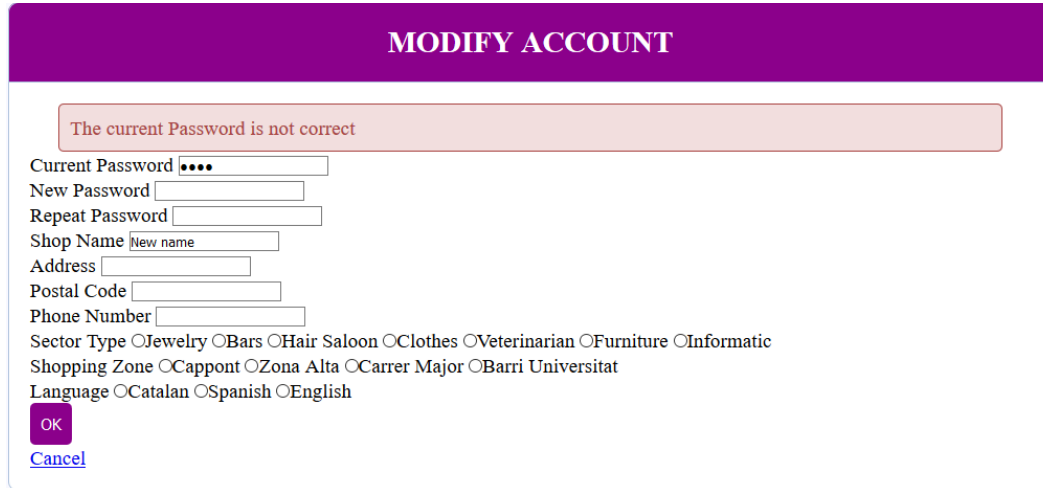


The screenshot shows a web page titled "Login successful" with a purple header. Below the header is a green success message box that says "You are now logged in". The page content includes the text "Welcome test@gmail.com" followed by two links: "continue" in green and "logout" in red.

Image 24 Login successful

MODIFY ACCOUNT

Imagine a user wants to modify some information about his/her account. So as to update the information, the current password of the account is required, so, if this password is not correctly, an error is displayed on the screen, such as this:

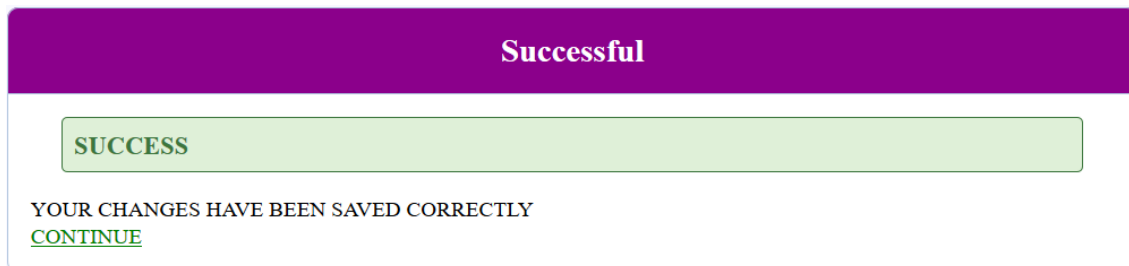


The screenshot shows a web form titled "MODIFY ACCOUNT" with a purple header. Below the header, a red error message box states "The current Password is not correct". The form contains several input fields: "Current Password" (with a masked password "...."), "New Password", "Repeat Password", "Shop Name" (with a placeholder "New name"), "Address", "Postal Code", and "Phone Number". Below these fields are several radio button options for "Sector Type" (Jewelry, Bars, Hair Saloon, Clothes, Veterinarian, Furniture, Informatic), "Shopping Zone" (Cappont, Zona Alta, Carrer Major, Barri Universitat), and "Language" (Catalan, Spanish, English). At the bottom of the form are two buttons: a purple "OK" button and a blue "Cancel" link.

Image 25 Modify Account

In case the current password field is not filled in, an error message is also displayed.

In case that everything is correct, a success page is displayed on the screen, saying that the changes have been uploaded into the database, here we can see it:

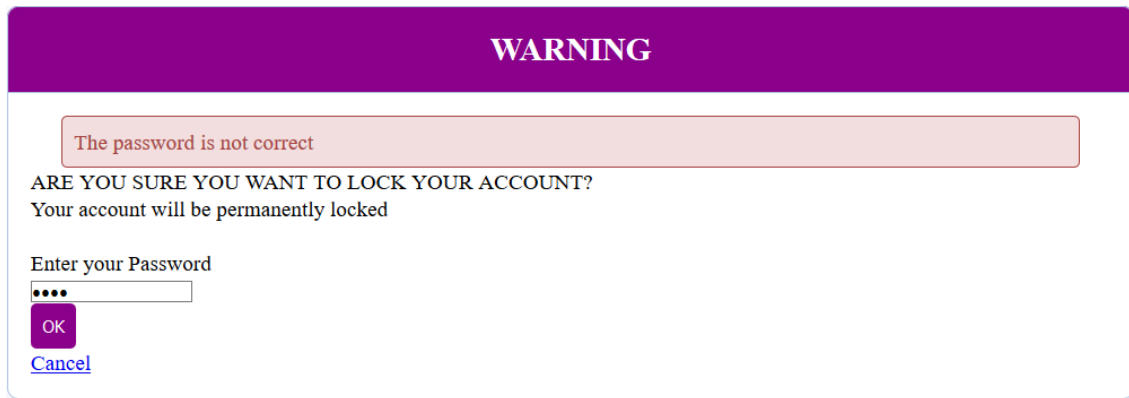


The screenshot shows a web page titled "Successful" with a purple header. Below the header, a green success message box states "SUCCESS". Below this, the text "YOUR CHANGES HAVE BEEN SAVED CORRECTLY" is displayed, followed by a green "CONTINUE" link.

Image 26 Successful

CLOSE/LOCK ACCOUNT

In the pages of locking or closing the account, the possible scenarios are the same in both. It is required to introduce the password so as to lock or close the account, so if this field is not filled or the password is not correct, an error message is displayed. Here we have the example:



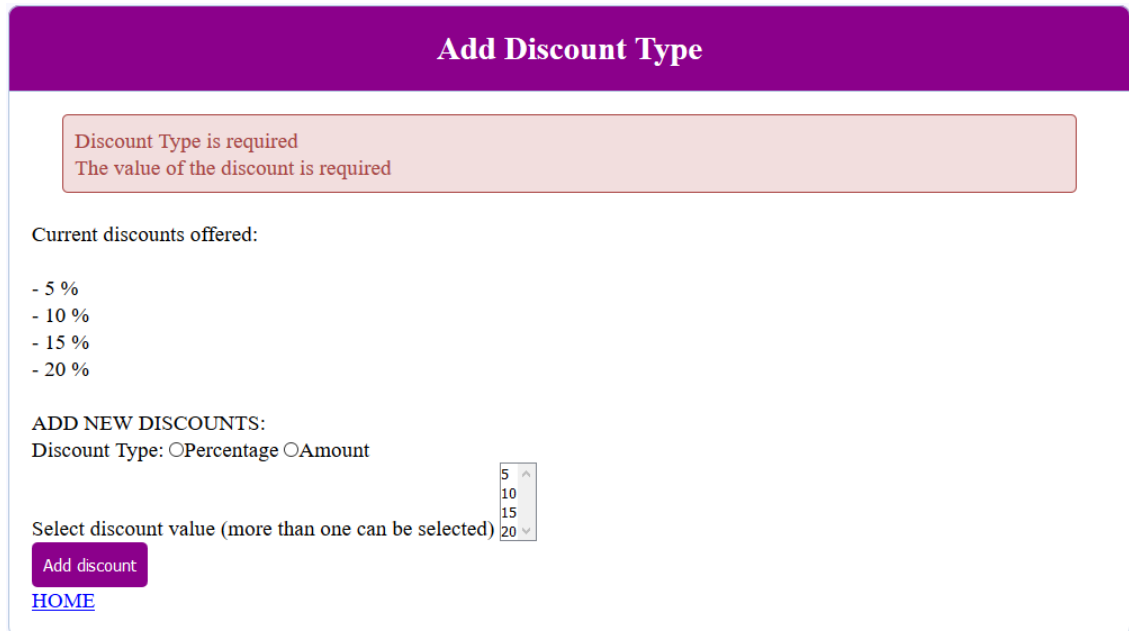
A warning dialog box with a purple header bar containing the word "WARNING" in white. Below the header, a red-bordered box contains the text "The password is not correct". Underneath, the text "ARE YOU SURE YOU WANT TO LOCK YOUR ACCOUNT?" is followed by "Your account will be permanently locked". A label "Enter your Password" is above a password input field showing four dots. At the bottom left are "OK" and "Cancel" buttons.

Image 27 Warning

In case that the password is correct, the account will be automatically closed or locked, and the user will be redirected to the login page.

ADD DISCOUNT TYPE

The functionality of this page is for when a shop wants to add a new kind of discount type. Here we have the scenario that the user does not select any kind of discount. In that case, an error is displayed on the screen like this:



The "Add Discount Type" form has a purple header bar with the title "Add Discount Type". A red-bordered box displays two error messages: "Discount Type is required" and "The value of the discount is required". Below this, the text "Current discounts offered:" is followed by a list: "- 5 %", "- 10 %", "- 15 %", and "- 20 %". The section "ADD NEW DISCOUNTS:" contains "Discount Type: ☐Percentage ☐Amount". A dropdown menu for "Select discount value (more than one can be selected)" shows options 5, 10, 15, and 20. At the bottom left are "Add discount" and "HOME" buttons.

Image 28 Add Discount Type

In case everything is correct, a successful page is shown:

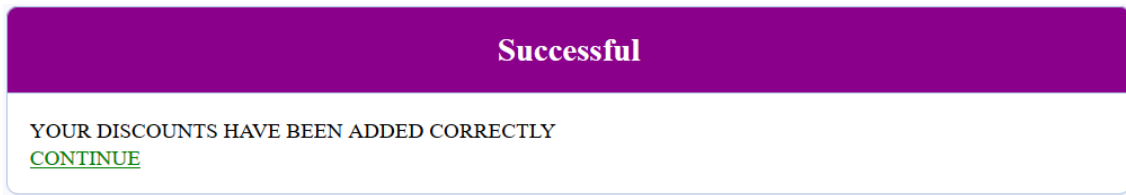


Image 29 Successful Page

ADD SECTOR TYPE

The functionality of this page is for when a customer wants to add a new kind of sector type. Here we have the scenario that the user does not select any kind of sector. In that case, an error is displayed on the screen like this:

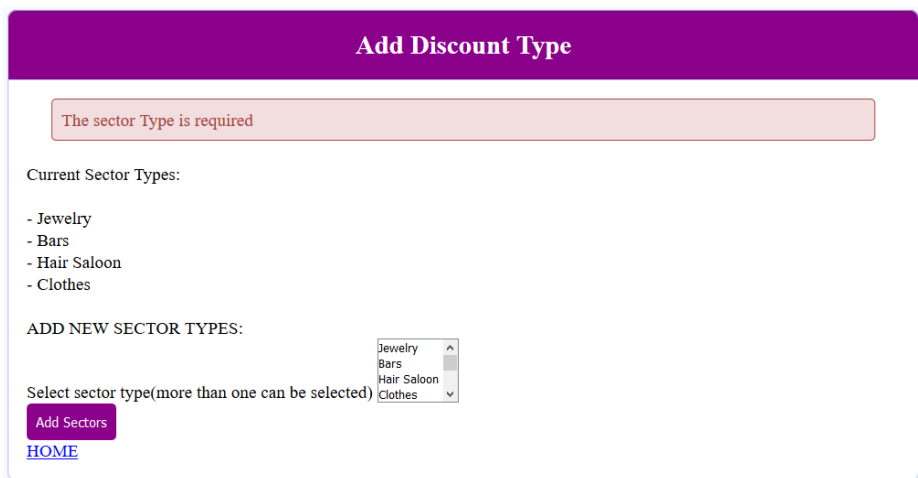


Image 30 Add Sector Type

In case everything is correct, a successful page is shown:

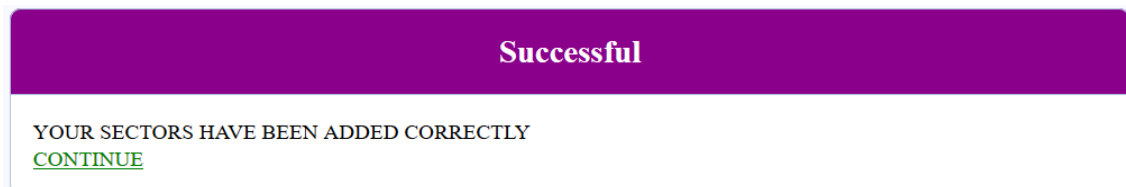


Image 31 Successful Page

9. FUTURE PROPOSALS

As we have said, this phase of the project is only the implementation of the web application. There some functionalities that have not been implemented, due to the size of the project. Now, we are going to explain in detail what could be the next steps to do in the whole project.

First of all, all the functionalities should be implemented. These are some of the functionalities:

- The admin page should be implemented. The admin has to control the accounts locked, and must be able to add sector types, shopping zones and discount types.
- According to the previous phase of the project, the application is thought to create alerts to the customer when he is located near a shop in which he has a voucher. This must be implemented via GPS.

In the first part of the project we have seen how to measure the customer loyalty. A proposal for the future so as to evaluate whether this application is useful for the shops or not, is to measure if this application has increased the customer loyalty.

The next step could be making the interface design of the application. By the moment, it is only functional and there is no design besides the application.

Finally, when the application runs perfect and the design is fine, the last step could be translating this web application into a mobile/tablet/computer application, so the shops can have them into one of these devices.

10. CONCLUSIONS

This project has been divided in two parts. The first part is related with the slope of business and the second part is related with the slope of engineering. First of all, we are going to explain the conclusions of every part of the project, and then, some global conclusions.

Referring to the business side, to sum up, we can say how important the loyalty of its customers is for a shop. Nowadays, the customer has an infinity variety of different shops in which to buy the same or similar product. Every day the customer is more sophisticated and has more knowledge of what he is doing, and the shops must offer a value added to the customers so as to encourage them to buy in the shop. We have seen that being careful with customers and trying to offer incentives such as discounts, is a good way to increase their loyalty. When doing this kind of projects such as creating a loyalty application, it is very important to measure if this application is value for the shops and customers and if it has done what it was pretended to do. So, it is very useful to do a post-analysis once the application is running on the market. This is the best way to measure if the customer loyalty has increased due to the application. Besides, doing a survey to know the opinion of the shops has been very useful when taking design decisions. We had problems when taking some decisions and the point of view of the shops have helped us to take those decisions. Doing a market analysis is always useful. Finally, doing this analysis of the customer loyalty and market has helped us to design the application and improve what it was already done.

On the other hand, in the engineering side, we could have checked all the functionality that the application has. By doing the implementation of the web application we could have fixed some errors committed during the first part. Doing this part is very useful to see whether this application works or not. The design of the database has been very useful so as to make a properly implementation of the application. We have found the best way to create the database, such as creating new tables so as to store the information needed for the application.

Now, with the contributions of this project, together with the first phase, the whole project of creating a loyalty application is very close to be finished. In this part we have seen how the application would work, and now it is the moment to finish implementing some functionalities and finally create the application for the correct device and launch it to the market. I think this project has been very useful to see a real prototype of how the application will work.

Doing this final grade project has been also very useful for me as a person and as a professional. I could have seen the complexity of a real project and the work that there is behind it. This project is only one part of the whole project, but I can figure out the full size of a real project. It has helped me to grow as a professional and I think that this is one of the most important things.

11. BIBLIOGRAPHY

Database Design with UML and SQL. (2018). Retrieved from <http://www.tomjewett.com/dbdesign/dbdesign.php?page=manymany.php>

Comer, D., & Stevens, D. (1993). Client-server programming and applications. Englewood Cliffs, New Jersey: Prentice-Hall.

Contributors, P. (2018). phpMyAdmin. Retrieved from: <https://www.phpmyadmin.net>

MySQL :: MySQL 8.0 Reference Manual :: 1 General Information. (2018). Retrieved from: <https://dev.mysql.com/doc/refman/8.0/en/introduction.html>

MySQL Tutorial - SQL Injection. (2018). Retrieved from <http://www.tizag.com/mysqlTutorial/mysql-php-sql-injection.php>

PHP: MySQL real escape string - Manual. (2018). Retrieved from <http://php.net/manual/en/function.mysql-real-escape-string.php>

PHP: History of PHP and Related Projects - Manual. (2018). Retrieved from <http://www.php.net/history>

PHP: History of PHP - Manual. (2018). Retrieved from <http://php.net/manual/en/history.php.php>

Ranade, K. (2012). Customer Loyalty - What is it? How Can You Measure and Manage It? - Loyalty Research Center. [Online] Retrieved from : <http://www.loyaltyresearch.com/insights/customer-loyalty-what-is-it-how-can-you-measure-and-manage-it/>

Reichheld, F., & Teal, T. (2001). *The loyalty effect*. Boston, Mass.: Harvard business school Press.

Rodriguez-del-Bosque Rodriguez, I., Herrero Crespo, A., and Garcia-de-los-Salmones Sanchez, M. (2008). *The influence of managers ' innovativeness on the implementation of loyalty programs in the small retailing sector*. Santander: Universidad de Cantabria.

What is customer loyalty? Definition and meaning. (2018). Retrieved from <http://www.businessdictionary.com/definition/customer-loyalty.html>